University of Alberta

Resistive Hybrid Tactile Sensor

by

Kam Chau So

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of **Master of Science**.

Department of Computing Science

Edmonton, Alberta
Spring 2000

National Library of Canada
Acquisitions and Bibliographic Services
395 Wellington Street
Ottawa ON K1A 0N4
Canada

Bibliothèque nationale du Canada
Acquisitions et services bibliographiques
395, rue Wellington
Ottawa ON K1A 0N4
Canada

Your file Votre référence

Our file Notre référence

The author has granted a non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-60187-0

Canada

University of Alberta

Library Release Form

Name of Author: Kam Chau So

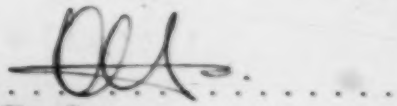
Title of Thesis: Resistive Hybrid Tactile Sensor

Degree: Master of Science

Year this Degree Granted: 2000

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as hereinbefore provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.



Kam Chau So
18839-51 Ave
Edmonton, Alberta
Canada, T6M 2L1

Date: March 20, 2000

University of Alberta

Faculty of Graduate Studies and Research

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled **Resistive Hybrid Tactile Sensor** submitted by Kam Chau So in partial fulfillment of the requirements for the degree of **Master of Science**.

Dr Hong Zhang

Dr Duncan Elliott

Dr Jim Hoover

Date: March 20, 2000

Abstract

This thesis introduces a new touch sensor design, called a hybrid tactile sensor. It is a combination of analog and digital touch sensing technologies. In practice, analog touch sensing technology is simple and widely used in electronic touch screens and touchpads. However, it is limited to detecting point contacts. A Digital touch sensing array, on the other hand, provides shape discrimination, but it requires more complex circuit design and generates larger amount of raw data for processing. This research focuses on the development of a new sensor design that is simple and capable of shape recognition. Different models of hybrid tactile sensors are studied and simulations are done to verify their characteristics. Physical prototypes are constructed using inexpensive materials, such as conductive silicone rubber and fabric mesh. Experiments are done and their results are analyzed.

To my wife, Mimi

Acknowledgements

Thanks to my supervisor, Dr Hong Zhang for all the support throughout the research. His guidance and numerous ideas are invaluable.

Thanks to the support from the people in the Computer Vision and Robotics Research Group. I am grateful to Johnathan who provides the skeleton of the communication program used to interface the cylindrical hybrid sensor to the terminal. Thanks also go to Dr Duncan Elliott and Dr Jim Hoover for their constructive suggestions.

Last but not least, thanks to my wife, Mimi for spending so much time proof-reading the manuscript of this thesis.

Contents

1 Introduction 1
- 1.1 Overview of Tactile Sensing Technologies and Sensors . . . 1
- 1.2 Motivation . . . 4
- 1.3 Thesis Outline . . . 5

2 Background and Related Information 6
- 2.1 Analog Tactile Sensing . . . 6
- 2.2 Digital Tactile Sensing . . . 8
- 2.3 Typical Characteristics of Tactile Sensors . . . 11

3 Design of The Hybrid Tactile Sensors and Their Modeling 12
- 3.1 Model One: Pairs of Conductive Rubber Strips (CRS-CRS) . . . 12
- 3.2 Model Two: Conductive Rubber Strips on Top of Metal Sheet (CRS-M) . . . 17
- 3.3 Model Three: Conductive Rubber Strips on Top of a Conductive Rubber Plane (CRS-CRP) . . . 18
- 3.4 Model Four: Cylindrical Variant CRS-M (Cyl-CRS-M) . . . 19

4 Shape Interpretation 22
- 4.1 Pseudo-Centers of Contact Shape . . . 22
  - 4.1.1 k_h-Pseudo-Center . . . 22
  - 4.1.2 k_l-Pseudo-Center . . . 23
  - 4.1.3 k-Pseudo-Center . . . 24
- 4.2 Vertex Reconstruction . . . 25
  - 4.2.1 Assumptions on Shapes . . . 25
  - 4.2.2 Characteristics of a Pseudo Signature . . . 26
  - 4.2.3 Reliability of a Pseudo-Signature . . . 27
  - 4.2.4 Recovering the Vertices . . . 33
- 4.3 Profile Reconstruction . . . 39
  - 4.3.1 Assumptions on Shapes . . . 39
  - 4.3.2 Recovering the Profile . . . 40
  - 4.3.3 Comparison Between Vertex Reconstruction and Profile Reconstruction 40

5 Simulations 42
- 5.1 Sources of Errors . . . 42
  - 5.1.1 Combined Quantization Noise . . . 42
  - 5.1.2 Error in Detecting Transitions of a Pseudo-signature . . . 43
  - 5.1.3 Positional Uncertainty . . . 43
- 5.2 Simulated Scenarios . . . 44

5.3 Metrics to Measure Shape Recovery Performance 46
5.4 Simulation Results with a CRS-CRS Hybrid Sensor 47
5.4.1 Vertex Reconstruction with a CRS-CRS Hybrid Sensor using k_h-Pseudo-Signature 47
5.4.2 Vertex Reconstruction with a CRS-CRS Hybrid Sensor using k_l-Pseudo-Signature 59
5.4.3 Profile Reconstruction with a CRS-CRS Hybrid Sensor 69
5.5 Simulation Results with a CRS-M Hybrid Sensor 81
5.6 Summary of Simulations 90

6 Experimental Results **92**
6.1 Experiments with a CRS-CRS Sensor 92
6.1.1 Calibrations of a CRS-CRS Sensor 93
6.1.2 Results of Recovering a Rectangle Using the Vertex Reconstruction Algorithm with a CRS-CRS Sensor and k_h-Pseudo-Signature 94
6.1.3 Results of Recovering a Rectangle Using the Vertex Reconstruction Algorithm with a CRS-CRS Sensor and k_l-Pseudo-Signature 102
6.1.4 Results of Recovering a Rectangle Using the Profile Reconstruction Algorithm with a CRS-CRS Sensor 106
6.2 Experiments with a CRS-M Sensor 109
6.2.1 Calibrations of a CRS-M Sensor 110
6.2.2 Results of Recovering a Rectangle Using the Vertex Reconstruction Algorithm with a CRS-M Sensor 113
6.3 Experiments with a Cyl-CRS-M Sensor 116
6.3.1 Calibrations of a Cyl-CRS-M Sensor 117
6.3.2 Results of Recovering Points/Lines Contacts with a Cly-CRS-M Sensor 118
6.4 Summary of Experiments 118

7 Conclusions **121**
7.1 Summary of Objectives 121
7.2 Summary of Simulations 122
7.3 Summary of Prototypes and Experiments 122
7.4 Summary of Achievement 123
7.5 Further Research 124

Bibliography **125**

List of Figures

2.1 A Typical Four-Wire Analog Resistive Tactile Sensor 7
2.2 A Typical Digital Tactile Sensor with $n \times n$ cross-switches 8
2.3 (a) Sensor Output of an Arbitrary Object from an Analog Tactile Sensor. (b) Sensor Output of the Same Object from an Digital Tactile Sensor. . . . 10
2.4 Possible Parallel Path in a Digital Tactile Sensor 10

3.1 A CRS-CRS Hybrid Tactile Sensor Model 13
3.2 Equivalent Circuit of a Pair of Rubber Strips in Contact with an Object at Position l with a Width w 14
3.3 k_h as a function of l and w. 15
3.4 k_l as a function of l and w. 15
3.5 A CRS-M Hybrid Tactile Sensor Model 17
3.6 Equivalent Circuit of a Rubber Strip in Contact with an Object at Position l with a Width w against a Metal Bottom Sheet 18
3.7 k as a function of l and w. 19
3.8 A CRS-CRP Hybrid Tactile Sensor Model 20
3.9 Equivalent Circuit of a Rubber Strip in Contact with an Object at Position l with a Width w against a Uniform Conductive Rubber Sheet 20
3.10 Using a Cyl-CRS-M Hybrid Tactile Sensor to Detect Point and Line Contacts 21

4.1 An Arbitrary Object on a Hybrid Tactile Sensor 23
4.2 The k_h-Pseudo-Signatures of An Arbitrary Four-sided Polygon at Different Location on a Hybrid Tactile Sensor. 24
4.3 The k_l-Pseudo-Signatures of An Arbitrary Four-sided Polygon at Different Location on a Hybrid Tactile Sensor. 25
4.4 The k-Pseudo-Signatures of An Arbitrary Four-sided Polygon at Different Location on a Hybrid Tactile Sensor. 26
4.5 Ambiguity due to concavity (a) multiple interpretations (two polygons satisfying the same set of k_h (dashed lines are $k_h \times L$)) (b) under-determinedness (four variables required to define the two contact regions) 27
4.6 Geometric Signature and Pseudo-Signature of an Arbitrary Four-sided Polygon 28
4.7 Signature Transition from One Section to Another with a Vertex at the Top 29
4.8 Signature Transition from One Section to Another with a Vertex at the Bottom 30
4.9 Geometric Signature and Pseudo-Signatures of an Arbitrary Tetragon . . . 35
4.10 An Arbitrary Pseudo-Signature with Three Blocks of Sections 37

5.1 Quantization at the Edge of Contact between an Object and the Hybrid Sensor. 43
5.2 Typical Examples that Constitute Positional Uncertainty 44

5.3 Turn Functions of Polygons A and B. 46
5.4 Mismatch Coefficient (M.C.) Between Polygons A and B shown in Figure 5.3. 47
5.5 Recovering a Triangle on an ideal CRS-CRS Sensor by V.R. with k_h-Pseudo-Signature 49
5.6 Recovering a Triangle on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_h-Pseudo-Signature 49
5.7 Recovering a Triangle on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_h-Pseudo-Signature 50
5.8 Recovering a Triangle on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. P.U. Profile II) by V.R. with k_h-Pseudo-Signature 50
5.9 Noise-free and Noisy k_h-Pseudo-Signature of Triangular Shape 5 with P.U. Profile II 51
5.10 Recovering a Tetragon on an ideal CRS-CRS Sensor by V.R. with k_h-Pseudo-Signature 52
5.11 Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_h-Pseudo-Signature 53
5.12 Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_h-Pseudo-Signature 53
5.13 Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_h-Pseudo-Signature 54
5.14 Noise-free and Noisy k_h-Pseudo-Signature of Tetragonal Shape 4 with P.U. Profile II 55
5.15 Recovering a Hexagon on an ideal CRS-CRS Sensor by V.R. with k_h-Pseudo-Signature 56
5.16 Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_h-Pseudo-Signature 56
5.17 Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_h-Pseudo-Signature 57
5.18 Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_h-Pseudo-Signature 57
5.19 Recovering a Concave Polygon on an ideal CRS-CRS Sensor Using V.R. with k_h-Pseudo-Signature 58
5.20 Recovering a Triangle on an ideal CRS-CRS Sensor by V.R. with k_l-Pseudo-Signature 59
5.21 Recovering a Triangle on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_l-Pseudo-Signature 60
5.22 Recovering a Triangle on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_l-Pseudo-Signature 60
5.23 Recovering a Triangle on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_l-Pseudo-Signature 61
5.24 Recovering a Tetragon on an ideal CRS-CRS Sensor by V.R. with k_l-Pseudo-Signature 62
5.25 Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_l-Pseudo-Signature 63
5.26 Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_l-Pseudo-Signature 63
5.27 Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_l-Pseudo-Signature 64

5.28 Recovering a Hexagon on an ideal CRS-CRS Sensor by V.R. with k_l-Pseudo-Signature 65
5.29 Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_l-Pseudo-Signature 66
5.30 Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_l-Pseudo-Signature 66
5.31 Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_l-Pseudo-Signature 67
5.32 Recovering a Concave Polygon on an ideal CRS-CRS Sensor Using V.R. with k_l-Pseudo-Signature 68
5.33 Recovering a Triangle on an ideal CRS-CRS Sensor by P.R. 69
5.34 Recovering a Triangle on a CRS-CRS Sensor (C.Q.E.) by P.R. 70
5.35 Recovering a Triangle on a CRS-CRS Sensor (C.Q.E. & P.U. Profile I) by P.R. 70
5.36 Recovering a Triangle on a CRS-CRS Sensor (C.Q.E. & P.U. Profile II) by P.R. 71
5.37 Recovering a Tetragon on an ideal CRS-CRS Sensor by P.R. 72
5.38 Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E.) by P.R. 73
5.39 Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E. & P.U. Profile I) by P.R. 73
5.40 Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E. & P.U. Profile II) by P.R. 74
5.41 Recovering a Hexagon on an ideal CRS-CRS Sensor by P.R. 75
5.42 Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E.) by P.R. 76
5.43 Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E. & P.U. Profile I) by P.R. 76
5.44 Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E. & P.U. Profile II) by P.R. 77
5.45 Recovering a Concave Tetragon on an ideal CRS-CRS Sensor by P.R. . . . 78
5.46 Recovering a Concave on a CRS-CRS Sensor (C.Q.E.) by P.R. 79
5.47 Recovering a Concave Tetragon on a CRS-CRS Sensor (C.Q.E & P.U. Profile I) by P.R. 79
5.48 Recovering a Concave Tetragon on a CRS-CRS Sensor (C.Q.E & P.U. Profile II) by P.R. 80
5.49 Recovering a Triangle on a ideal CRS-M Sensor by V.R. 81
5.50 Recovering a Triangle on a CRS-M Sensor (C.Q.E. & T.E.) by V.R. 82
5.51 Recovering a Triangle on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. 82
5.52 Recovering a Triangle on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. 83
5.53 Recovering a Tetragon on a ideal CRS-M Sensor by V.R. 84
5.54 Recovering a Tetragon on a CRS-M Sensor (C.Q.E. & T.E.) by V.R. 85
5.55 Recovering a Tetragon on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. 85
5.56 Recovering a Tetragon on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. 86
5.57 Recovering a Hexagon on a ideal CRS-M Sensor by V.R. 87
5.58 Recovering a Hexagon on a CRS-M Sensor (C.Q.E. & T.E.) by V.R. 88
5.59 Recovering a Hexagon on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. 88

5.60 Recovering a Hexagon on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. 89
5.61 Recovering a Concave Polygon on an ideal CRS-M Sensor Using V.R. with k-Pseudo-Signature 90

6.1 Prototype of a CRS-CRS Sensor with Ten Sensing Elements 93
6.2 Calibration Graphs for the First Sensing Element in the CRS-CRS Hybrid Sensor Prototype ($w = 0$). 95
6.3 Calibration Graphs for the First Sensing Element in the CRS-CRS Hybrid Sensor Prototype ($w = 5$). 96
6.4 Calibration Graphs for the First Sensing Element in the CRS-CRS Hybrid Sensor Prototype ($w = 10$). 97
6.5 Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Lower Region of the CRS-CRS Sensor (with Calibration). 99
6.6 Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Lower Region of the CRS-CRS Sensor (without Calibration). 99
6.7 Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Middle Region of the CRS-CRS Sensor (with Calibration). 100
6.8 Noisy k_h-Signature of a Rectangle in the Middle Region of the CRS-CRS Sensor (without Calibration). 100
6.9 Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Upper Region of the CRS-CRS Sensor (with Calibration). 101
6.10 Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Upper Region of the CRS-CRS Sensor (without Calibration). 101
6.11 Recovering a Rectangle Using V.R. Algorithm with k_l-Signature in the Lower Region of the CRS-CRS Sensor (with Calibration). 102
6.12 Noisy k_l-Signature of a Rectangle in the Lower Region of the CRS-CRS Sensor (without Calibration). 103
6.13 Recovering a Rectangle Using V.R. Algorithm with k_l-Signature in the Middle Region of the CRS-CRS Sensor (with Calibration). 103
6.14 Noisy k_l-Signature of a Rectangle in the Middle Region of the CRS-CRS Sensor (without Calibration). 104
6.15 Recovering a Rectangle Using V.R. Algorithm with k_l-Signature in the Upper Region of the CRS-CRS Sensor (with Calibration). 104
6.16 Noisy k_l-Signature of a Rectangle in the Upper Region of the CRS-CRS Sensor (without Calibration). 105
6.17 Recovering a Rectangle Using P.R. Algorithm in the Lower Region of the CRS-CRS Sensor (with Calibration). 106
6.18 Recovering a Rectangle Using P.R. Algorithm in the Lower Region of the CRS-CRS Sensor (without Calibration). 107
6.19 Recovering a Rectangle Using P.R. Algorithm in the Middle Region of the CRS-CRS Sensor (with Calibration). 107
6.20 Recovering a Rectangle Using P.R. Algorithm in the Middle Region of the CRS-CRS Sensor (without Calibration). 108
6.21 Recovering a Rectangle Using P.R. Algorithm in the Upper Region of the CRS-CRS Sensor (with Calibration). 108
6.22 Recovering a Rectangle Using P.R. Algorithm in the Upper Region of the CRS-CRS Sensor (without Calibration). 109

6.23 Prototype of a CRS-M Sensor with Nine Sensing Elements 110
6.24 Calibration Graphs for the First Sensing Element in the CRS-M Hybrid Sensor Prototype ($w = 0$). 111
6.25 Calibration Graphs for the First Sensing Element in the CRS-M Hybrid Sensor Prototype ($w = 5$). 111
6.26 Calibration Graphs for the First Sensing Element in the CRS-M Hybrid Sensor Prototype ($w = 10$). 112
6.27 Recovering a Rectangle Using V.R. Algorithm in the Lower Region of the CRS-M Sensor (with Calibration). 113
6.28 Noisy k_l-Signature of a Rectangle in the Lower Region of the CRS-M Sensor (without Calibration). 114
6.29 Recovering a Rectangle Using V.R. Algorithm in the Middle Region of the CRS-M Sensor (with Calibration). 114
6.30 Recovering a Rectangle Using V.R. Algorithm in the Middle Region of the CRS-M Sensor (without Calibration). 115
6.31 Recovering a Rectangle Using V.R. Algorithm in the Upper Region of the CRS-M Sensor (with Calibration). 115
6.32 Recovering a Rectangle Using V.R. Algorithm in the Upper Region of the CRS-M Sensor (without Calibration). 116
6.33 Prototype of a Cyl-CRS-M Sensor with Nine Sensing Elements 117
6.34 Calibration Graphs for the First Sensing Element in the Cyl-CRS-M Hybrid Sensor Prototype ($w = 0$). 118
6.35 Recovering Points/Lines Contacts Using the CRS-M Sensor. 119

List of Tables

2.1 Suggested Tactile Sensor Requirements 11

4.1 Block Types 39
4.2 Comparison Between Vertex Reconstruction and Profile Reconstruction . . 40

5.1 Simulated Error Profiles for Positional Uncertainty 45
5.2 Quantitative Mismatch between the Original and the Recovered Triangle in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_h-Pseudo-Signature). 51
5.3 Quantitative Mismatch between the Original and the Recovered Tetragon in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_h-Pseudo-Signature). 54
5.4 Quantitative Mismatch between the Original and the Recovered Hexagon in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_h-Pseudo-Signature). 58
5.5 Quantitative Mismatch between the Original and the Recovered Triangle in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_l-Pseudo-Signature). 61
5.6 Quantitative Mismatch between the Original and the Recovered Tetragon in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_l-Pseudo-Signature). 64
5.7 Quantitative Mismatch between the Original and the Recovered Hexagon in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_l-Pseudo-Signature). 68
5.8 Quantitative Mismatch between the Original and the Recovered Triangle in Each Simulated Position (CRS-CRS Sensor and P.R. Algorithm). 71
5.9 Quantitative Mismatch between the Original and the Recovered Tetragon in Each Simulated Position (CRS-CRS Sensor and P.R. Algorithm). 74
5.10 Quantitative Mismatch between the Original and the Recovered Hexagon in Each Simulated Position (CRS-CRS Sensor and P.R. Algorithm). 77
5.11 Quantitative Mismatch between the Original and the Recovered Concave Tetragon in Each Simulated Position (CRS-CRS Sensor and P.R. Algorithm). 80
5.12 Quantitative Mismatch between the Original and the Recovered Triangle in Each Simulated Position (CRS-M Sensor and V.R. Algorithm). 83
5.13 Quantitative Mismatch between the Original and the Recovered Tetragon in Each Simulated Position (CRS-M Sensor and V.R. Algorithm). 86
5.14 Quantitative Mismatch between the Original and the Recovered Hexagon in Each Simulated Position (CRS-M Sensor and V.R. Algorithm). 89

6.1 Calibration Factors for Each Sensing Element of the CRS-CRS Hybrid Sensor Prototype. 94
6.2 Quantitative Mismatch Between the Original and the Recovered Rectangle in Each Experiment (CRS-CRS Sensor Prototype and V.R. Algorithm with k_h-Signature). 98
6.3 Quantitative Mismatch Between the Original and the Recovered Rectangle in Each Experiment (CRS-CRS Sensor Prototype and V.R. Algorithm with k_l-Signature). 105
6.4 Quantitative Mismatch Between the Original and the Recovered Rectangle in Each Experiment (CRS-CRS Sensor Prototype and P.R. Algorithm). . . 106
6.5 Calibration Factors for Each Sensing Element of the CRS-M Hybrid Sensor Prototype. 112
6.6 Quantitative Mismatch Between the Original and the Recovered Rectangle in Each Experiment (CRS-M Sensor Prototype and V.R. Algorithm with k-Signature). 116
6.7 Calibration Factors for Each Sensing Element of the Cyl-CRS-M Hybrid Sensor Prototype. 117
6.8 Percentage Errors in V_h w.r.t. V_{ref} for Each Sensing Element of the CRS-CRS Hybrid Sensor Prototype in the Experiments (after Calibration). . . . 119
6.9 Percentage Errors in V_l w.r.t. V_{ref} for Each Sensing Element of the CRS-CRS Hybrid Sensor Prototype in the Experiments (after Calibration). 120
6.10 Percentage Errors in V_{out} w.r.t. V_{ref} for Each Sensing Element of the CRS-M Hybrid Sensor Prototype in the Experiments (after Calibration). 120
6.11 Percentage Errors in V_{out} w.r.t. V_{ref} for Each Sensing Element of the Cyl-CRS-M Hybrid Sensor Prototype in the Experiments (after Calibration). . . 120

Chapter 1

Introduction

The sense of touch is an important sensing modality in robotic applications. Various tactile information such as location, orientation, size, shape, weight, pressure, temperature and surface texture are important for a robot to navigate and perform its task efficiently and safely [1] [2]. To collect tactile information, a tactile sensor is used. It measures the parameters of contact interaction between the device and a physical stimulus. In general, the interaction takes place within a touch-sensitive region of the device's surface. Although limited by the structure, materials and applications of a tactile sensor, it is possible for its sensing surface to conform to virtually any shape or contour [3]. An overview of tactile sensing technologies and sensors is presented in the next section. It is followed by an introduction on the motivation of the research. An outline of the thesis will be given in the last section.

1.1 Overview of Tactile Sensing Technologies and Sensors

Tactile sensors can be categorized in a number of ways. For instance, they are commonly categorized by the way they transduce tactile parameters into a form suitable for computer analysis [4]. The following is a summary of tactile sensors commonly used in robotic applications:

(a) Resistive and conductive tactile sensors measure the change in resistance or conductance of a resistive or conductive material when an external force is applied. The material commonly used is elastomer (elastic polymer). Its deformation by an external pressure causes a change in local particle density, which in turns alters its resistivity or conductivity. This type of tactile sensors are among the earliest being developed and widely used in industry. These sensors have a large dynamic range, good overload tolerance and are very robust. However, they also have problems such

as hysteresis, limited spatial resolution, non-linear response characteristics. Typical designs are those developed by Robertson and Walkden [5] and M. Inaba *et al.* [6].

(b) Piezoelectric tactile sensors generate voltage across the sensing element when pressure is exerted on them, whereas pyroelectric tactile sensors generate a voltage when there is a change of temperature to the sensing element. The generated voltage is proportional to the applied pressure for the former and temperature for the latter. In both types of sensors, no external voltage is required and continuous analog output is available. They are inherently dynamic. If pressure or temperature is applied and kept unchanged, the corresponding sensor output decays to zero. In other words, these sensors are most suitable for registering the change in pressure or temperature. The materials commonly used are polymers with piezoelectric or pyroelectric properties. These sensors usually have a wide dynamic range. They are quite durable and capable of either force or temperature measurement. However, they have drawbacks such as complexity in sensor designs. Moreover, the piezoelectric and pyroelectric effects generally co-exist in the sensor materials, and hinder the applications when only either one of the parameters is required. Typical examples of these types of sensors are designed by Dario and De Rossi [7], J. S. Son *et al.* [8] and Dargahi *et al.* [9].

(c) Capacitive tactile sensors measure capacitance that varies with applied loads. The sensor element consists of an dielectric elastomer sandwiched between the electrode plates of a capacitor. External force is applied and causes physical deformation or displacement of the elastomer that changes the effective capacitance of the sensing element. Such a change in capacitance is proportional to the external pressure. These sensors are fairly robust, have a wide dynamic range and a reasonably linear response. However, they are prone to noise, sensitive to temperature change and have limited spatial resolution. A typical example of this kind of sensors is described by Siegel *et al.* [10].

(d) Magnetic tactile sensors detect a change in magnetic field by either of the following ways. First, magnetoelastic materials are used, which exhibit a change in the magnetic field when subjected to mechanical stress. Second, it is possible to detect the change of magnetic flux from relative mechanical displacement between the core and the surrounding coil of a solenoid-like structure. Those sensors based on mechanical displacement are simple in design and exhibit a wide dynamic range. Yet their spatial resolution is limited, whereas sensors that use magnetoelastic materials are relatively

superior in detecting normal force, torque and shear force. However, they are prone to noise, and additional AC noise shielding circuitry is required to protect the materials.

(e) Mechanical tactile sensors measure the mechanical displacement caused by an applied force. A typical example is a linear potentiometer with its sliding arm (output electrode) moved by an external force to produce an output proportional to the displacement (or the applied force). Even a spring-loaded switch that indicates an open or close contact is a common sensing element. In general, mechanical sensors are simple and robust. They can be used to sense the magnitude of applied force, linear and angular displacements. However, they are usually bulky and provide very limited spatial resolution.

(f) Electrochemical tactile sensors are based on the phenomenon of streaming potentials. In the example proposed by De Rossi *et al.* [11], an ionized gel containing an immobile negative charge with balanced amount of mobile positive charge is used. The gel is contained in a compliant structure. When external force is applied to the gel, positively charged liquid is forced out of the gel into a reservoir. Thus, an imbalanced amount of ions in the gel constitutes a streaming potential which can be used to indicate the magnitude of pressure applied.

(g) Optical tactile sensors use optomechanical transduction coupled with mechanical displacement. A typical sensor is made up of an array of sensing elements covered by a compliant surface. Each sensing element has an elongated pin aligned with a photoemitter/detector pair. When an external force is applied on the compliant surface, affected pins will be displaced inward and block the corresponding light paths of its photoemitter/detector pair. The amount of displacement indicates the amount of light reaching the photodetector that, when digitized, represents the amount of pressure applied on the sensor surface. In practice, optical sensors offer very high spatial resolution. They are also inherently immune from electrical interference and can be easily integrated with other vision-based sensing modalities. However, their designs are relatively complex and costly.

(h) Acoustic tactile sensors are based on the resonant frequency of ultra-sound. In the setup used by H. Shinoda *et al.* [12], a sensing element is made up of a cavity in an elastic body from which two paths are extended to an ultrasonic transmitter/receiver pair. With no external force applied, the acoustic impedance of the cavity is high enough to block the transmission path between the transmitter/receiver pair. When

external force is exerted on the elastic body and distorted the shape of the cavity, resonant frequencies can be picked up by the receiver. Thus, the type of deformation made by a point, a line or a plane can be determined by its relationship corresponding to the resonant frequencies. Another design using a PVDF ultrasonic array is described by B. L. Hutchings *et al.* [13].

Alternatively, tactile sensors can also be classified by their dimensionalities as follows:

(a) A *zero-dimensional sensor* is the basic sensor that detects point contacts.

(b) A *one-dimensional sensor* is a collinear arrangement of zero-dimensional sensors.

(c) In a *two-dimensional sensor* , the sensing elements are arranged as a grid of dimensions $m \times n$, where m and n are the number of sensing elements in the horizontal and vertical dimensions respectively.

1.2 Motivation

Though tremendous progress is found in tactile sensing research, there are still considerable problems in existing tactile sensor designs. First, most of the tactile sensors are small in size as they are designed to fit in a dexterous robot hand. Hence they are not suitable for use on large surfaces such as the links of a robot arm or the body of a humanoid robot. Second, all existing tactile sensors are expensive to fabricate. Their costs range from $500 to $1000 per square inch. It would be extremely expensive for applications requiring large sensing areas. Third, analog tactile sensor is the simplest in design and easiest to fabricate; however, it is only capable of detecting point-contacts. No matter what the shape of the object is, only the "*geometric average* " of the area of contact can be detected and such information is represented by a pair of co-ordinates relative to the sensor's reference frame. Fourth, although digital sensing is very common and provides more precise information, such as the profile, location, and orientation of the shape in contact, the amount of raw data is on the order of n^2 for an $n \times n$ digital sensing array. It could impose considerable software processing overhead when n is large. For instance, if $n = 1000$ and each sensing element provides a single byte of data, a million bytes of data are generated in each sampling for further processing. Fifth, there is a "*parallel path*" problem [14] associated with certain designs of digital sensing array. Special circuitry is required to eliminate any phantom footprint of the original shape, which means additional cost in the manufacturing process. Finally, the existing tactile sensors tend to have fixed geometries (planar, spherical, and cylindrical) and are difficult for adaptation into arbitrary shapes.

This thesis focuses on improvements of the above-mentioned inadequacies by compromising between the analog and the digital[1] extremes of sensing techniques. Resistive material is one of the most inexpensive and commonly used elements in constructing a tactile sensor. In this study, the classical analog resistive sensor architecture is used as the basic sensing element. A hybrid sensor prototype by integrating a number of these sensing elements is introduced. This prototype shows that it is possible to put together low cost materials and relatively simple technology to improve over the inadequacies of either the analog or the digital sensors.

1.3 Thesis Outline

The rest of the thesis consists of six chapters. Chapter two provides background information in analog tactile sensing and resistive tactile sensors. Chapter three presents four models of hybrid tactile sensors. Chapter four explains two shape interpretation algorithms that work with the hybrid tactile sensors. Chapter five presents the simulation and its results of a hypothetical sensor and the shape interpretation algorithms. Chapter six discusses the experimental results of three hybrid tactile sensor prototypes. Chapter seven concludes the thesis and suggests directions for future research.

[1]It would be more proper to use the terms "continuous" and "discrete" instead of "analog" and "digital" respectively. However, the latters are widely used in the robotics industry. In this thesis, the terms "analog" and "digital" are adapted as synonyms for "continuous" and "discrete" respectively.

Chapter 2

Background and Related Information

In this chapter, an overview will be given on both analog tactile sensing and digital tactile sensing technologies. Representative models for each of them will be explained and their associated merits and inadequacies will be discussed. Finally, the last section of this chapter presents the desirable characteristics of a tactile sensor.

2.1 Analog Tactile Sensing

Analog touch sensing has been widely used in the computer industry [15]. For instance, touchpads are standard pointing devices in notebook or palm-top computers, whereas electronic whiteboards can be found in many classrooms and conference centers. Common to all these applications is a touch-based input device. Various transduction technologies have been used in achieving this capability, including capacitive, force/strain gauge, acoustic, scanning infrared, and analog resistive. In theory, most of them are zero-dimensional and categorized as analog tactile sensors. For these devices to work, one critical assumption is that the contact with the touch-sensitive surface occurs only at a point, so that the associated analog technologies can resolve the position of the contact.

Of particular interest to this research is the analog resistive technology. One common configuration of an analog resistive touch screen is the so-called *4-wire* technology, as shown in Figure 2.1. It consists of two conductive sheets with a finite resistance, typically on the order of 100 ohms per square inch, placed on top of each other. These conductive sheets are normally separated from each other when there is no contact. The top sheet carries a voltage gradient generated by applying a reference voltage on one end of the sheet and ground at the other end. The second sheet serves as the slider in a linear potentiometer. When contact

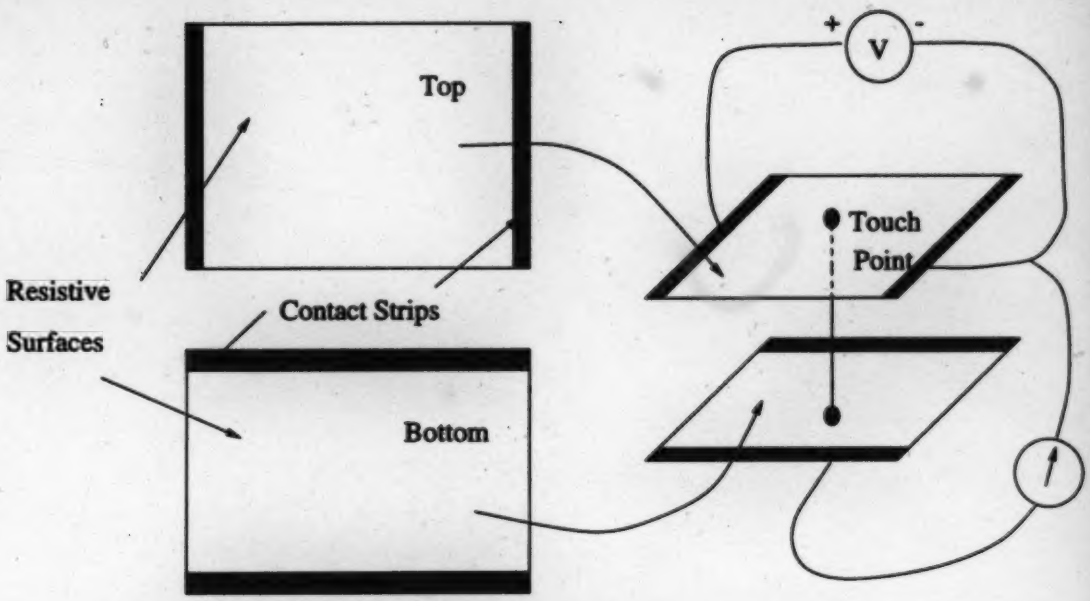


Figure 2.1: A Typical Four-Wire Analog Resistive Tactile Sensor

is made, the sheets touch at a point, creating a voltage divider. The position of the contact can be determined from the voltage measured by the bottom sheet. By reversing the roles of the sheets and repeating the above process, one can obtain a second measurement. These measurements can be used to resolve the x-y coordinates of the contact point. The sampling of the two voltages and the subsequent calculation in this analog resistive sensor are both extremely simple. The complexity of the solution does not increase with the size of the sensor surface. The sensor itself can be produced with inexpensive materials.

A number of techniques can be used to produce the spacing between the two conductive sheets. One solution is to use an air gap between the two sheets. This is a particularly simple solution for the case of a planar sensor surface such as an electronic whiteboard. Another solution is to use microspheres, which are small pressure-sensitive dots whose diameters can be on the order of one tenth of a millimeter. Yet another solution is to use an insulating fabric mesh between the two sheets. The critical pressure causing the contact can be controlled by the thickness of the fabric and the size of the holes of the mesh.

Analog resistive technology, and analog technologies in general, are an effective approach to solving the electronic touch screen problem. However, this approach is not directly applicable to robot tactile sensing because of its inability to detect multiple points of contact or determine the shape of contact. For instance, if the contact occurs at two points, the volt-

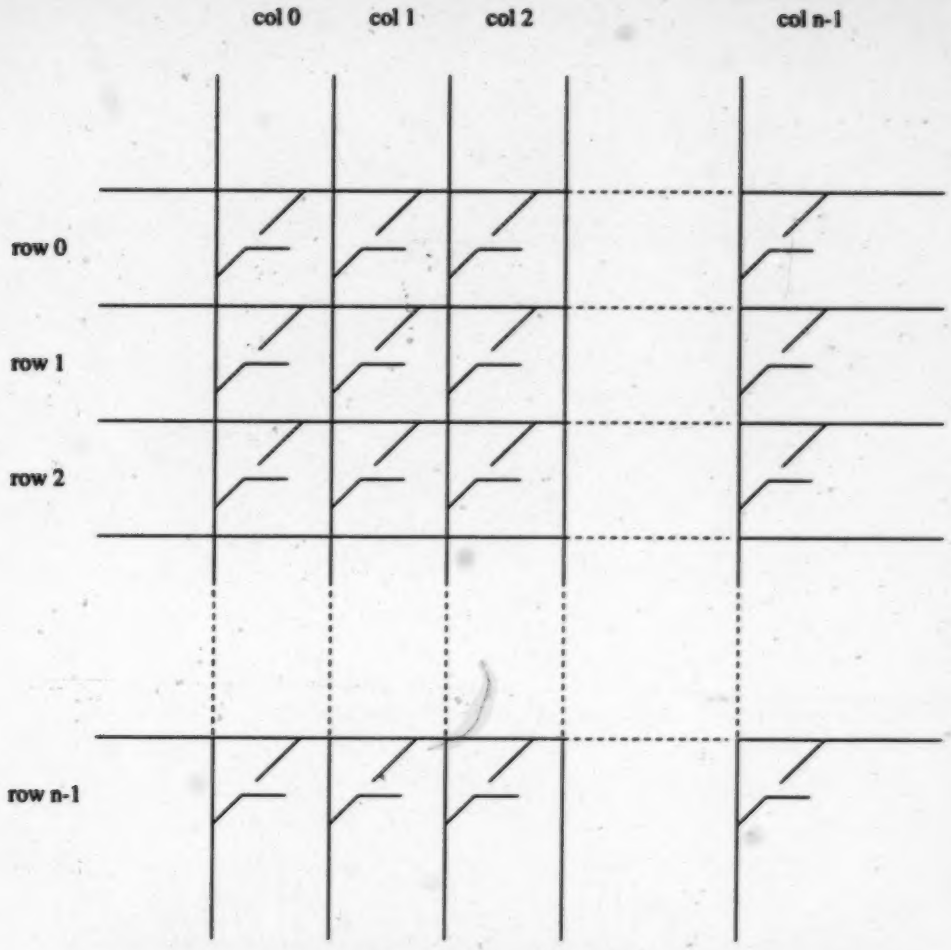


Figure 2.2: A Typical Digital Tactile Sensor with $n \times n$ cross-switches

age measurements will indicate the a point somewhere between the two contact positions. Therefore, design changes must be introduced in order to take advantage of the simplicity of the analog touch technologies.

2.2 Digital Tactile Sensing

Digital tactile sensing uses a two-dimensional array of sensing elements, called *tactels* (analogous to pixels for picture elements). Common applications of digital tactile sensors can be found on touch screens of cash registers in supermarkets, on vending machines at street corners, and on control panels of advanced equipment. Assuming there are n tactels in each row and column, the sensors area is digitized into $n \times n$ sensing sites. The simplest arrange-

ment can be a matrix of on/off switches, which indicates contact or no contact condition at the corresponding site. Figure 2.2 illustrates a typical digital tactile sensor made up of $n \times n$ cross-switches. The status of each cross-switch is obtained by the following operations:

Step 1: Apply reference signal to one and only one row, starting from row 0.

Step 2: Scan all columns, starting from column 0 to $n-1$ and record the presence of any reference signal in each column. Presence of a reference signal at any column x indicates a closure of cross-switch at row 0 and column x.

Step 3: Repeat step one and two with reference signal applied to row 1 and so on, up to row $n-1$.

With the same sensing surface area, a digital sensor offers n^2 as many sensing elements as its analog counterpart. As a result, a digital sensor provides a much higher spatial resolution. The states of the tactels constitute a tactile image of the object in contact. Such an image is far superior than just the single-point contact information provided by an analog sensor. Figure 2.3 illustrates the advantage of a digital sensor output over an analog sensor. Information of a contact object, such as its location, area, shape and orientation can be readily captured by a digital tactile sensor.

Though a digital tactile sensor is superior to its analog counterpart, the grid-like arrangement of tactels in a digital sensor causes a "parallel path" problem, as illustrated in Figure 2.4. Assume that only cross-switches $(i+1,j)$, $(i+1,j+1)$ and $(i,j+1)$ are closed. When a reference signal is applied to row i, it will also appear at the output of column j due to the existence of a parallel path made up of cross-switches $(i+1,j)$, $(i+1,j+1)$ and $(i,j+1)$. Hence, cross-switch (i,j) will be mistakenly recorded as "closed".

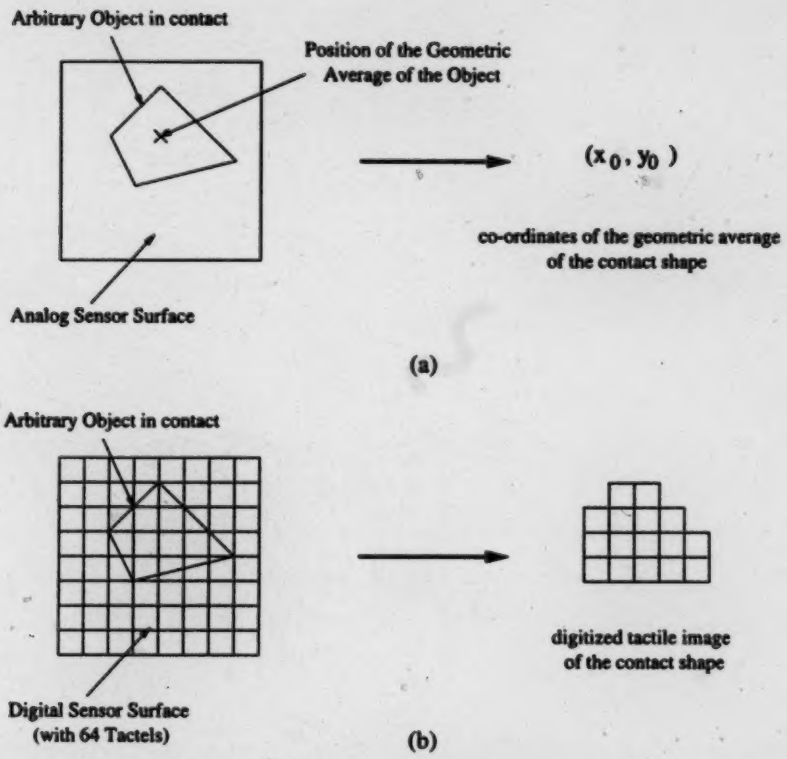

Arbitrary Object in contact
Position of the Geometric Average of the Object
(x_0, y_0)
co-ordinates of the geometric average of the contact shape
Analog Sensor Surface
(a)
Arbitrary Object in contact
digitized tactile image of the contact shape
Digital Sensor Surface (with 64 Tactels)
(b)

Figure 2.3: (a) Sensor Output of an Arbitrary Object from an Analog Tactile Sensor. (b) Sensor Output of the Same Object from an Digital Tactile Sensor.

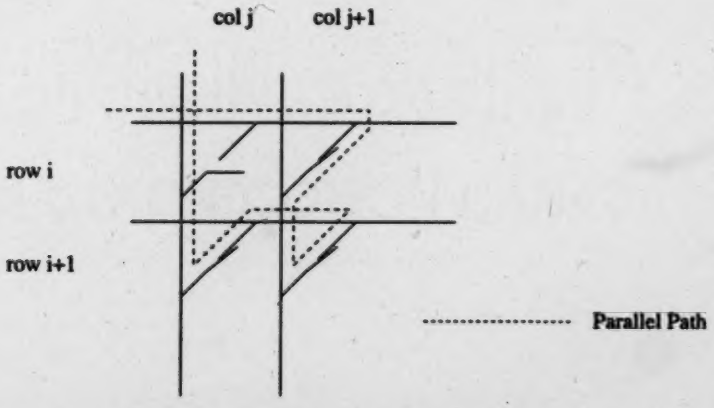


Figure 2.4: Possible Parallel Path in a Digital Tactile Sensor

Sensor Quality	Requirement
Sensor Surface	Should be both compliant and durable.
Spatial Resolution	Should be $1 - 2mm$.
Number of Sensing Elements	Should be $50 - 200$.
Minimum Sensitivity	Should be able to detect as little as $5g$, or ideally $1g$.
Sensor Output	Must be stable, repeatable and without hysteresis.
Output Response	Must be monotonic but not necessarily linear.
Sampling Frequency	Should be higher than $100Hz$.
Dynamic Range	Should be $1000 : 1$.

Table 2.1: Suggested Tactile Sensor Requirements

2.3 Typical Characteristics of Tactile Sensors

A survey done by L. D. Harmon [16] suggests that no matter what form of transduction is used, a set of tactile sensor requirements is desirable. Table 2.1 gives a summary of these requirements. Though the survey was done in the 1980's, it is still widely adopted for new tactile sensor designs.

Chapter 3

Design of The Hybrid Tactile Sensors and Their Modeling

Strictly speaking, this research exploits the capability of a one-dimensional sensor. The prototypes are called hybrid touch sensors that resulted from the combination of the analog resistive and the array touch sensing technologies. Based on various materials as well as structural and geometrical arrangements, four different sensor designs and their analytical models are presented in this chapter.

3.1 Model One: Pairs of Conductive Rubber Strips (CRS-CRS)

The first model is shown in Figure 3.1. Similar to a typical analog resistive sensor, there are two conductive sheets, one on top of the other. These sheets are made of conductive rubber, a form of elastomer doped with a conductive material. By using materials of different conductivity and/or controlling the amount of conductive materials in the manufacturing process, one can produce conductive rubbers with different conductivities. In the model, each conductive rubber sheet is divided into multiple columns or strips which are parallel to each other. Each strip on the top sheet is aligned to a corresponding strip on the bottom sheet. Effectively, there is a one dimensional array of analog resistive sensing elements, each behaving in a similar fashion to the analog touch sensor described in Section 2.1. During operation, a reference voltage is applied at one end of a strip on the top sheet and output is measured at the end of the corresponding strip on the bottom sheet. This process is done on each pair of strips in turn to complete a sampling cycle. As a result, the sensor measures the "average" position of contact along each strip and is able to detect contacts that occur along different columns. Since the sensor has a *Conductive Rubber Strip* on *Conductive*

Rubber Strip structure, it is called a *CRS-CRS* sensor.

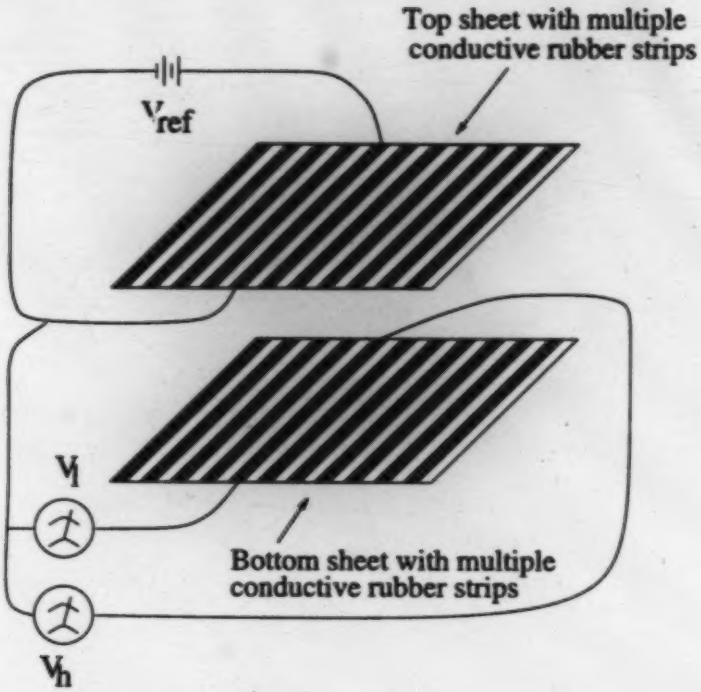


Figure 3.1: A CRS-CRS Hybrid Tactile Sensor Model

The equivalent of one pair of the sensor strips in Figure 3.1 is shown in Figure 3.2, together with its circuit diagram. The two sheets are assumed to have an identical conductivity. It is also assumed that the object contacts the sensor strip continuously so that only two parameters are required to describe the contact: the lower coordinate of the contact area (l) and the width of the contact area (w). Let L be the total length of a rubber strip in a sensing element, ρ be its linear resistivity and V_{ref} be the reference voltage. The measured voltages, V_h (output voltage at upper measuring point) and V_l (output voltage at lower measuring point) are given by:

$$\begin{aligned} V_h &= \frac{l\rho + \frac{w\rho}{2}}{L\rho - w\rho - l\rho + \frac{w\rho}{2} + l\rho} V_{ref} \\ &= \frac{l + w/2}{L - w/2} V_{ref} \end{aligned} \tag{3.1}$$

$$V_l = \frac{l\rho}{L\rho - w\rho - l\rho + \frac{w\rho}{2} + l\rho} V_{ref}$$

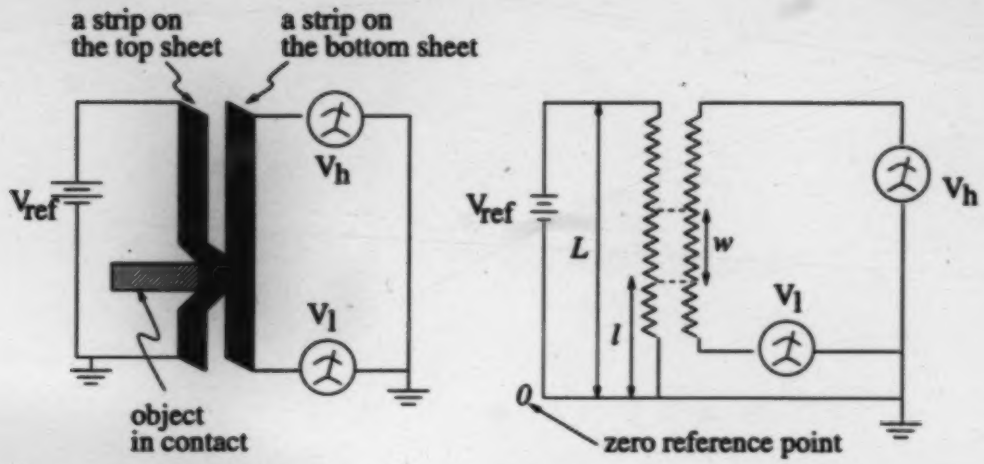


Figure 3.2: Equivalent Circuit of a Pair of Rubber Strips in Contact with an Object at Position l with a Width w

$$= \frac{l}{L - w/2} V_{ref} \tag{3.2}$$

Two quantities, k_h and k_l are further defined as follows:

$$k_h = \frac{V_h}{V_{ref}} = \frac{l + w/2}{L - w/2} \tag{3.3}$$

$$k_l = \frac{V_l}{V_{ref}} = \frac{l}{L - w/2} \tag{3.4}$$

Figure 3.3 and Figure 3.4 show the graphs of k_h and k_l (as functions of l and w) respectively. Since the quantities, L, l and w are bound by the physical dimensions of the hybrid sensor, the following constraints hold:

$$l \leq L, \qquad w \leq L \qquad \text{and} \qquad l + w \leq L$$

Since the sensor is discretized in the horizontal direction, it is a hybrid between a purely digital and a purely analog sensor. With n pairs of strips, if both V_h and V_l are measured,

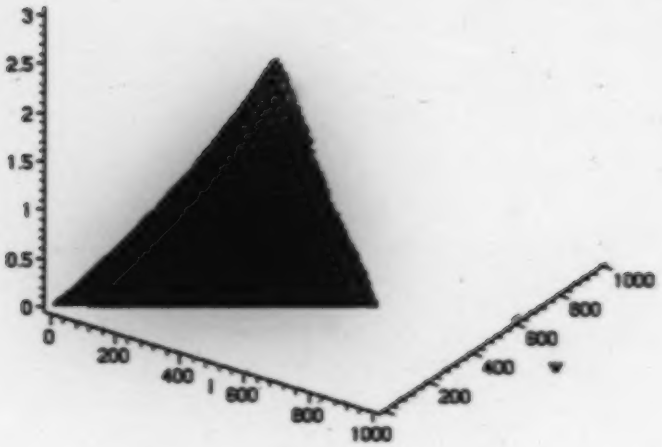


Figure 3.3: k_A as a function of l and w.

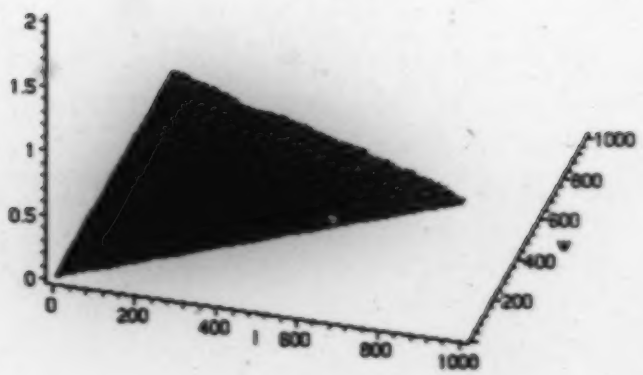


Figure 3.4: k_I as a function of l and w.

each sampling produces $2n$ data. It is defined as *double-measurement*. If only V_h or V_l is measured during a sampling cycle, only n data are available. It is defined as *single-measurement*. Two algorithms are derived to reconstruct the contact shape according to the output data set.

Each sensor required two connectors for V_{ref} and common ground. Each sensing element requires contacts for V_h and V_l. If there are n sensing elements in this type of sensor and if double-measurement is done, n connectors will be required for each V_h and V_l. Hence $2n+2$ connectors are required. If single-measurement is used, n connectors are needed for either V_h or V_l. So only $n+2$ connectors will be sufficient.

According to the definitions of k_h and k_l, the resistivities of the two rubber strips of a basic sensing element must be identical. In practice, such requirement can be relaxed. As long as their resistivities are known and linear along the length of each strip, then calibration can always be done. It should be noted that the definitions of k_h and k_l become:

$$k'_h = \frac{V'_h}{V_{ref}} = \frac{\rho_0 l + \alpha w}{\rho_0(L-l-w) + \alpha w + \rho_0 l} \tag{3.5}$$

$$k'_l = \frac{V'_l}{V_{ref}} = \frac{\rho_0 l}{\rho_0(L-l-w) + \alpha w + \rho_0 l} \tag{3.6}$$

where ρ_0 = resistivity of the energized rubber strip

ρ_1 = resistivity of the voltage measurement strip

$\alpha = \frac{\rho_0 \rho_1}{\rho_0 + \rho_1}$

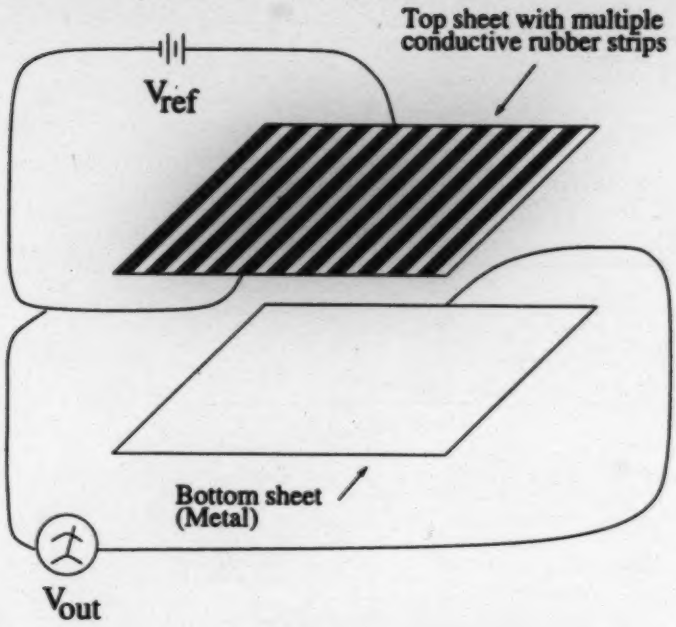


Figure 3.5: A CRS-M Hybrid Tactile Sensor Model

3.2 Model Two: Conductive Rubber Strips on Top of Metal Sheet (CRS-M)

The second model (see Figure 3.5) is a variation of the first configuration described in Section 3.1. There are also two sheets of conductive materials, one on top of the other. However, the bottom sheet is made of metal instead of conductive rubber. Similar energizing and measuring sequence is used. The parameters l and w are as defined in Section 3.1. As the bottom sheet is assumed to be a perfect conductor, the length of contact w of the sensor strip is practically a short circuit. V_h and V_l are virtually the same. Therefore only single-measurement (a set of n measurements on V_{out}) can be obtained in a sampling cycle. Since the sensor has a *Conductive Rubber Strip* on *Metal Sheet* structure, it is called a *CRS-M* sensor.

The equivalent circuit of one of the sensor strips against the metal sheet of Figure 3.5 is shown in Figure 3.6. As the length of contact w of the sensor strip is practically a short circuit, the measured voltage is given by:

$$V_{out} = \frac{l\rho}{(L-w-l)\rho + l\rho} V_{ref}$$

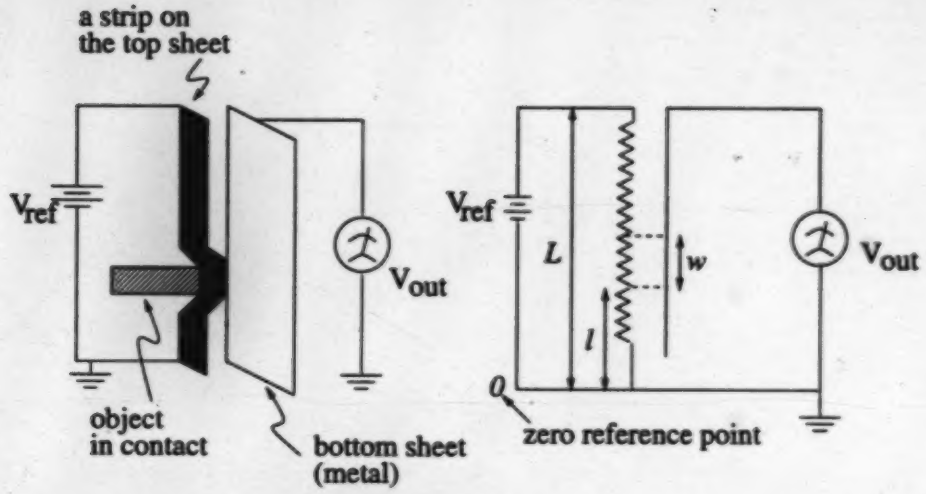


Figure 3.6: Equivalent Circuit of a Rubber Strip in Contact with an Object at Position l with a Width w against a Metal Bottom Sheet

$$= \frac{l}{L-w} V_{ref} \tag{3.7}$$

where L is the total length and ρ is the resistivity of the sensor strip respectively. The quantity, k is further defined as follows:

$$k = \frac{V_{out}}{V_{ref}} = \frac{l}{L-w} \tag{3.8}$$

Figure 3.7 shows the graph of k as functions of l and w. This sensor requires $2n+2$ connectors for n sensing elements. The resistivity of individual elastomer strip is not critical. However, it must be linear along the length of strip.

3.3 Model Three: Conductive Rubber Strips on Top of a Conductive Rubber Plane (CRS-CRP)

The third model (see Figure 3.8) is similar to the second model described in Section 3.2. The only difference is that the metal sheet at the bottom is replaced by a uniform conductive rubber sheet. In addition, it is assumed that the resistivity of the uniform conductive rubber

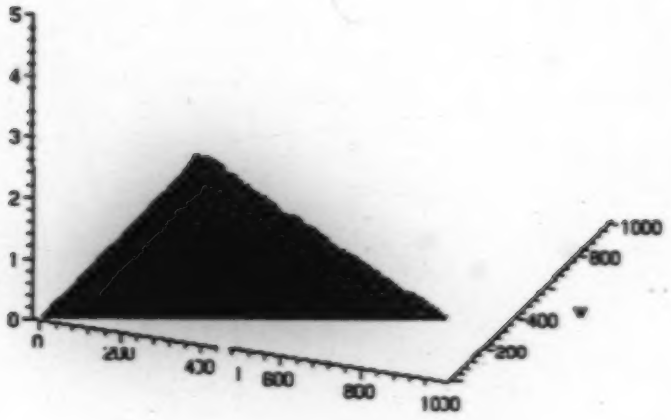


Figure 3.7: k as a function of l and w.

(ρ_{sheet}) is much higher tl an that of the top sheet (ρ_{strip}). Since the sensor has a *Conductive Rubber Strip* on *Uniform Conductive Rubber Sheet* structure, it is called a *CRS-CRP* sensor.

In each sampling cycle, reference voltage is applied to each strip of the top sheet in turn. Output voltages V_h and/or V_l are measured from the bottom sheet. In general, either single- or double-measurement can be taken from the sensor.

The equivalent circuit of one of the sensor strips against the uniform conductive rubber sheet of Figure 3.8 is shown in Figure 3.9. Unfortunately, the coupling resistance varies with different l and w. Hence V_l and V_h cannot be precisely expressed as simple mathematical functions of l and w. Only empirical data can be collected from different inputs of l and w. As a result, this setup is not suitable for practical use.

3.4 Model Four: Cylindrical Variant CRS-M (Cyl-CRS-M)

The fourth model is a geometrical variant of model two described in Section 3.2, so it is called a *Cyl-CRS-M* sensor. Instead of a planar geometry, the sensor is built into a cylindrical shape and resembles a robotic finger. The parameters and equations derived in Section 3.2 are also applicable to this sensor. Due to its relatively small curvature in

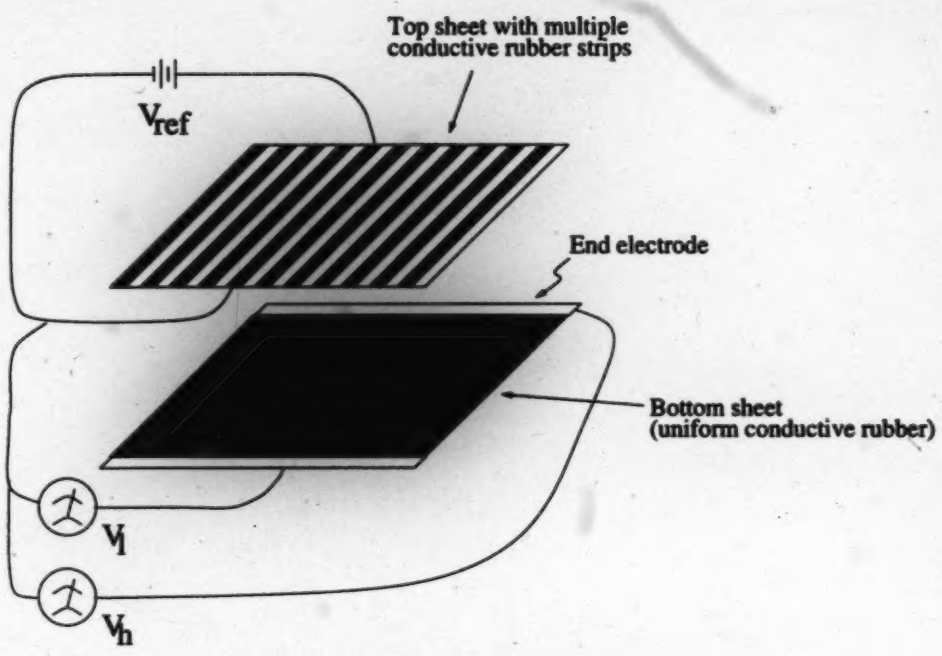


Figure 3.8: A CRS-CRP Hybrid Tactile Sensor Model

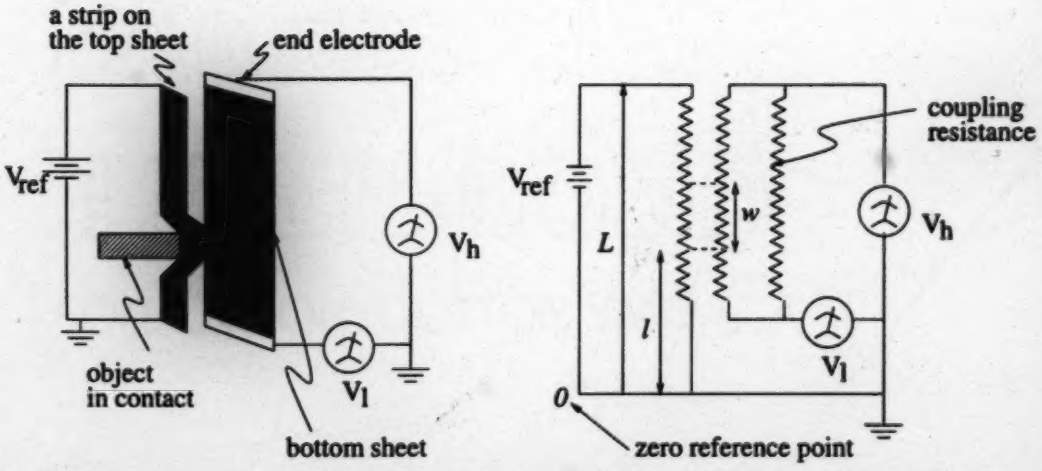


Figure 3.9: Equivalent Circuit of a Rubber Strip in Contact with an Object at Position l with a Width w against a Uniform Conductive Rubber Sheet

sensing surface, only point and limited linear contacts can be practically detected. Figure 3.10 illustrates such limitation. It can be seen from the cross-section view that only a few sensing strips can be in touch with the object. The output provided by the sensor is sufficient to determine the contact points and hence the portion of the object's edge in contact. However, no meaningful planar information about the object can be reconstructed.

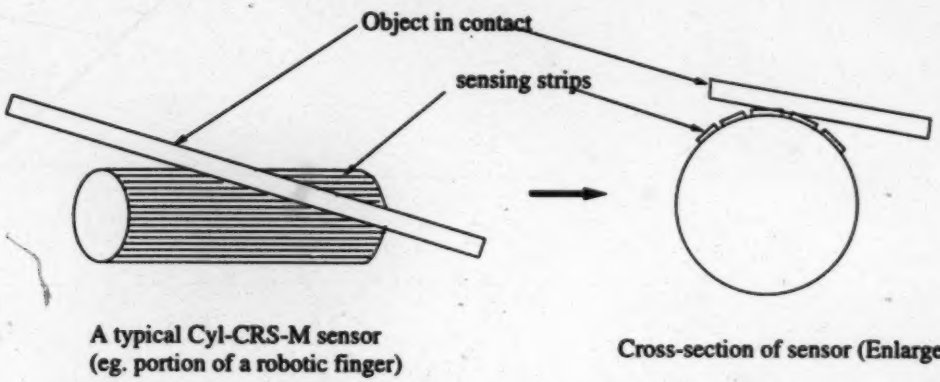

Object in contact
sensing strips
A typical Cyl-CRS-M sensor
(eg. portion of a robotic finger)
Cross-section of sensor (Enlarged View)

Figure 3.10: Using a Cyl-CRS-M Hybrid Tactile Sensor to Detect Point and Line Contacts

Chapter 4

Shape Interpretation

The outputs from the sensor models discussed in Chapter 3 provide abstracted geometric information about the contact shape. In order to recover geometric information, two shape interpretation algorithms, *Vertex Reconstruction* and *Profile Reconstruction* are derived according to the approach of output measurement. When applying either of these shape interpretation algorithms, some assumptions and limitations have to be considered.

4.1 Pseudo-Centers of Contact Shape

Figure 4.1 shows an arbitrary object on a hybrid tactile sensor. The quantity, $l + w/2$ from each strip is defined as the *geometric center* of contact. By joining all the geometric centers of contact, a *geometric signature* of the shape is obtained. Based on the values of k_h, k_l and k, three types of *pseudo-centers* of contact are defined.

4.1.1 k_h-Pseudo-Center

The quantity, $k_h L$ is defined as the *k_h-pseudo-center* of contact. Based on Equation (3.3), the k_h-pseudo-center differs from the geometric center by a factor of $\frac{L}{L - w/2}$, as shown in Equation (4.1). A *k_h-pseudo-signature* is obtained by joining all the k_h-pseudo-centers of contact.

$$k_h L = (l + \frac{w}{2}) \times \frac{L}{L - \frac{w}{2}} \tag{4.1}$$

It is noted that $\frac{L}{L - w/2} \geq 1$. Hence $k_h L$ is always above the geometric center in the reference frame of the sensor. In terms of transitional points, the profile of the pseudo

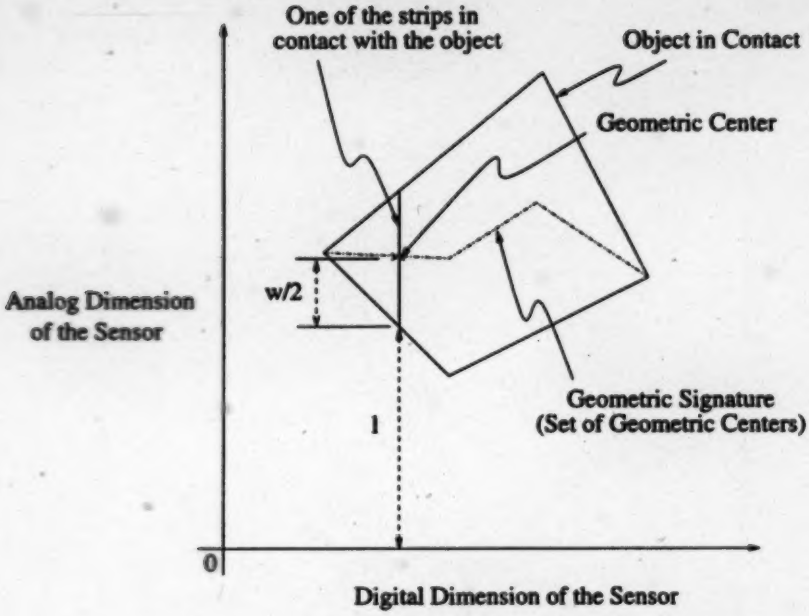


Figure 4.1: An Arbitrary Object on a Hybrid Tactile Sensor

signature follows that of the geometric signature closely. Figure 4.2 shows how the profile of a k_h-pseudo-signature changes as the contact shape moves across the sensor surface.

4.1.2 k_l-Pseudo-Center

The quantity, $k_l L$ is defined as the *k_l-pseudo-center* of contact. A *k_l-pseudo-signature* is obtained by joining all the k_l-pseudo-centers of contact. Equation (4.2) is readily obtained from the definition of k_l in Equation (3.4). Since $0 \leq k_l \leq 1$, the k_l-pseudo-center is always below the geometric center. It can be noted that as k_l approaches 0, $k_l L$ approaches l. For a practical range of w, the value of k_l is small when the contact shape is in the lower region of the sensor reference frame. Hence the profile of the k_l-pseudo-signature approaches that of the lower half of the contact shape. On the other hand, larger values of k_l occurs when the contact shape is in the upper region of the sensor. In terms of transitional points, the profile of the k_l-pseudo-signature closely resembles that of the geometric signature. Figure 4.3 illustrates how the profile of a k_l-pseudo-signature changes with the location of the contact shape on the sensor surface.

$$k_l L = l + k_l \frac{w}{2} \tag{4.2}$$

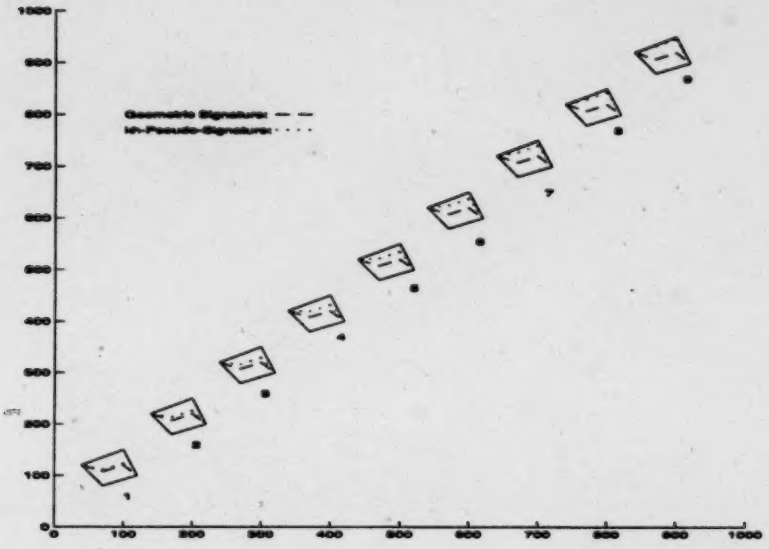


Figure 4.2: The k_h-Pseudo-Signatures of An Arbitrary Four-sided Polygon at Different Location on a Hybrid Tactile Sensor.

4.1.3 k-Pseudo-Center

The quantity, kL is defined as the *k-pseudo-center* of contact in the case of CRS-M model. A *k-pseudo-signature* is obtained by joining all the k-pseudo-centers of contact. Based on Equation (3.8), Equation (4.3) expresses kL in terms of l and w. It can be noted that the k-pseudo-center is above the geometric center when $k > \frac{1}{2}$, and vice versa when $k < \frac{1}{2}$. Since $0 \leq k \leq 1$, as k approaches 0, k_L approaches l. For a practical range of w, the values of k are small when the contact shape is in the lower region of the sensor reference frame. Hence the profile of the k-pseudo-signature approaches that of the lower half of the contact shape. On the contrary, when the contact shape is in the upper region of the sensor, larger values of k occur. The profile of the k-pseudo-signature tends to follow that of the upper half of the contact shape. When the contact shape is in the middle region of the sensor, the values of k are closer to $\frac{1}{2}$. In terms of transitional points, the profile of the k-pseudo-signature closely resembles that of the geometric signature. The changes in the profile of a k-pseudo-signature with respect to its location on the sensor is shown in Figure 4.4.

$$kL = l + kw \tag{4.3}$$

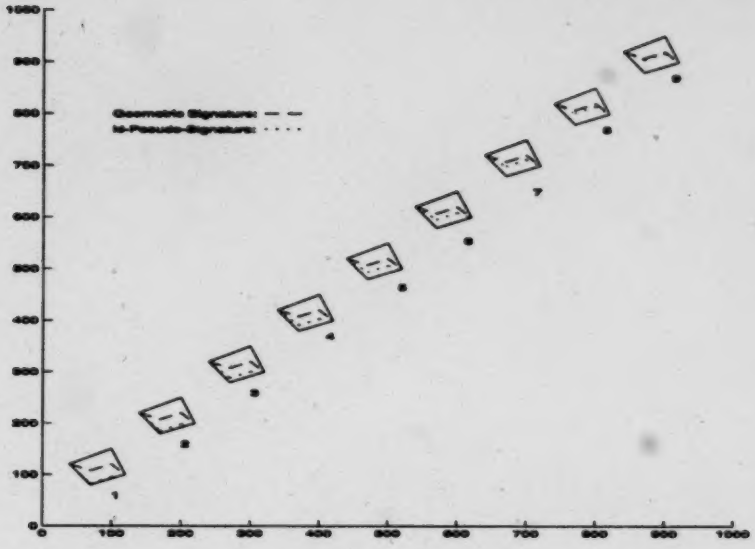


Figure 4.3: The k_l-Pseudo-Signatures of An Arbitrary Four-sided Polygon at Different Location on a Hybrid Tactile Sensor.

4.2 Vertex Reconstruction

Vertex Reconstruction is used when single-measurement is taken. It accepts an array of either k_h, k_l or k to recover the vertices of the corresponding contact shape. Before discussing how the algorithm works, a number of assumptions and limitations must be considered.

4.2.1 Assumptions on Shapes

First, it is assumed that the shape in contact must be polygonal so that it can be described by a set of vertices. Second, the polygonal shape must be strictly convex. Without such assumptions, a number of ambiguities arise. Figure 4.5 illustrates three possible cases. It can be noted that both shape X and Y in Figure 4.5(a) produce the same pseudo-signature. Whereas in Figure 4.5(b), the concavity of the shape Z makes it impossible to use just variables l and w to abstract the geometric information on the shaded portion of the shape. It is impossible to regenerate the contact shape uniquely from a given pseudo signature unless the convexity constraint is satisfied.

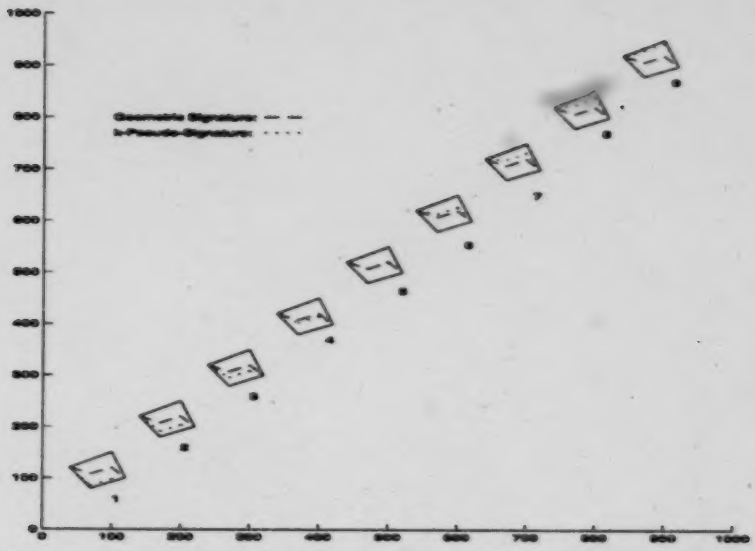


Figure 4.4: The k-Pseudo-Signatures of An Arbitrary Four-sided Polygon at Different Location on a Hybrid Tactile Sensor.

4.2.2 Characteristics of a Pseudo Signature

For simplicity, all different types of pseudo-signatures discussed in the Section 4.1 are called "pseudo-signature". Moreover, the terms "signature" and "pseudo-signature" are used interchangeably. It is important to make several observations on the characteristics of a pseudo-signature. First, the pseudo-centers of contact form a set of $N-1$ piece-wise continuous segments for an N-sided polygon if no two vertices share the same x coordinate. Figure 4.6 shows an example of a four-sided polygon. Excluding the two end vertices, there are two interior vertices. The whole polygon can be divided into three vertical sections by drawing a vertical line at each interior vertex. Since a pseudo-signature is a function of l and w, and it is piece-wise continuous in each section. At each interior vertex, either l or w changes to another linear function and discontinuity occurs. This discontinuity is reflected as transitional point on the pseudo-signature. In general, this is true in the case of a N-sided polygon. Excluding the two end vertices, there are $N-2$ interior vertices. The polygon can be divided into $N-1$ vertical sections by drawing a vertical line at each interior vertex. Hence there are $N-1$ piece-wise continuous segments in the corresponding pseudo-signature. However, when two vertices share the same x coordinate, a N-sided

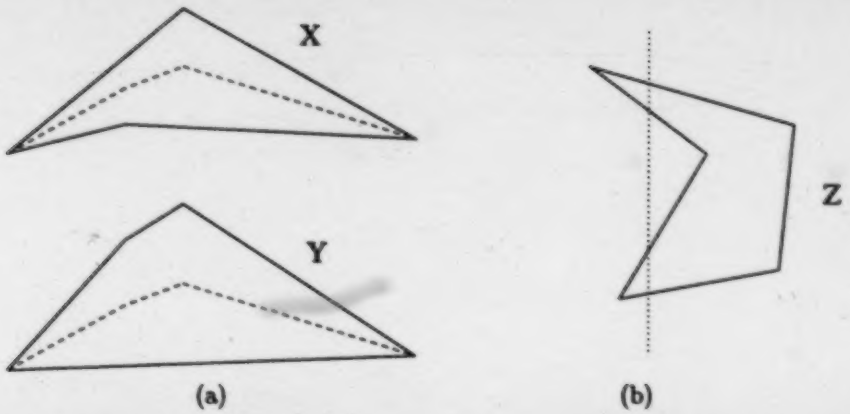


Figure 4.5: Ambiguity due to concavity (a) multiple interpretations (two polygons satisfying the same set of k_h (dashed lines are $k_h \times L$)) (b) under-determinedness (four variables required to define the two contact regions)

polygon produces a pseudo-signature with $N-2$ segments. This situation is considered as singular and one cannot apply the algorithm for shape recovery.

Second, the x coordinates of the vertices are found by detecting where line segments intersect. Along the pseudo-signature from left to right, at the intersection of two signature segments, there is a vertex above the intersection if the slope of the right line segment decreases. Otherwise, the vertex is below the intersection.

Finally, the end vertices can be determined directly from the pseudo-signature by setting $w = 0$ for these two vertices. The only unknowns of the system are the y coordinates of the interior vertices. Since equations (3.3), (3.4) and (3.8) hold for the corresponding sensor columns of these interior vertices, the polygonal shape can be solved exactly by setting up a linear system of $N-2$ equations with $N-2$ unknowns.

4.2.3 Reliability of a Pseudo-Signature.

As mentioned in Section 4.1, there are three types of pseudo-signatures based on k_h, k_l and k respectively. Though they have transitional points corresponding to the vertices of the polygonal shape, the accuracy in detecting them from the pseudo-signature determines how accurate the vertices can be recovered. In general, there is a discontinuity in the first-derivatives when two signature segments meet, except for a few configurations. With discontinuity in the first-derivatives at the intersection of two signature segments, it is relatively easy to locate the transitional point by detecting a prominent change in the gradient from one signature segment to another. Though there are a few configurations

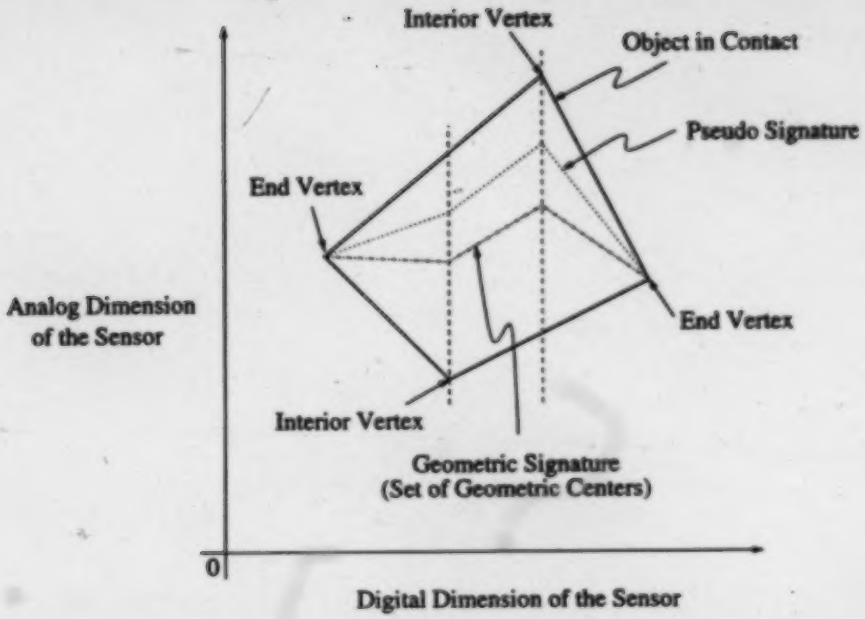


Figure 4.6: Geometric Signature and Pseudo-Signature of an Arbitrary Four-sided Polygon

that such discontinuity vanishes, they exist only when precise conditions are met. These conditions are illustrated with reference to Figure 4.7 and Figure 4.8, which depict all the possible circumstances.

In general, the following equations hold:

$$l_i = m_i x + c_i \qquad \text{for } i = 1, 2, \ldots, 6$$

where m_i and c_i are the slope and intercept of the corresponding linear equation for l_i $(i = 1, 2, \ldots, 6)$. Then the following equations also hold:

$$w_1 = l_2 - l_1 = (m_2 - m_1)x + c_2 - c_1 \tag{4.4}$$

$$w_2 = l_3 - l_1 = (m_3 - m_1)x + c_3 - c_1 \tag{4.5}$$

$$w_3 = l_6 - l_4 = (m_6 - m_4)x + c_6 - c_4 \tag{4.6}$$

$$w_4 = l_6 - l_5 = (m_6 - m_5)x + c_6 - c_5 \tag{4.7}$$

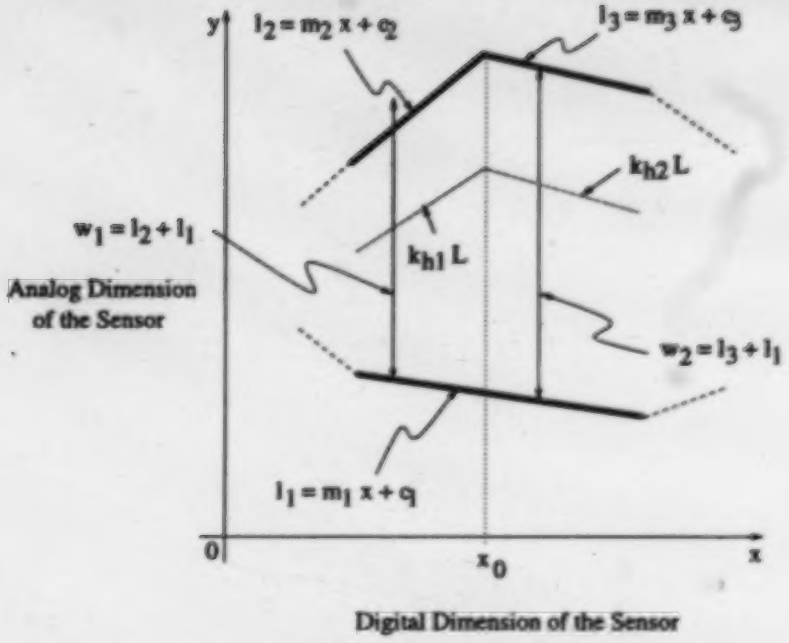


Figure 4.7: Signature Transition from One Section to Another with a Vertex at the Top

1. In case of k_h, the signature segments are given by

$$k_{h1}L = \frac{l_1 + \frac{w_1}{2}}{L - \frac{w_1}{2}} \times L = \frac{L(m_1x + c_1 + m_4x + c_4)}{2L - m_2x - c_2 + m_1x + c_1} \tag{4.8}$$

$$k_{h2}L = \frac{l_1 + \frac{w_2}{2}}{L - \frac{w_2}{2}} \times L = \frac{L(m_1x + c_1 + m_3x + c_3)}{2L - m_3x - c_3 + m_1x + c_1} \tag{4.9}$$

$$k_{h3}L = \frac{l_4 + \frac{w_3}{2}}{L - \frac{w_3}{2}} \times L = \frac{L(m_4x + c_4 + m_6x + c_6)}{2L - m_6x - c_6 + m_4x + c_4} \tag{4.10}$$

$$k_{h4}L = \frac{l_5 + \frac{w_4}{2}}{L - \frac{w_4}{2}} \times L = \frac{L(m_5x + c_5 + m_6x + c_6)}{2L - m_6x - c_6 + m_5x + c_5} \tag{4.11}$$

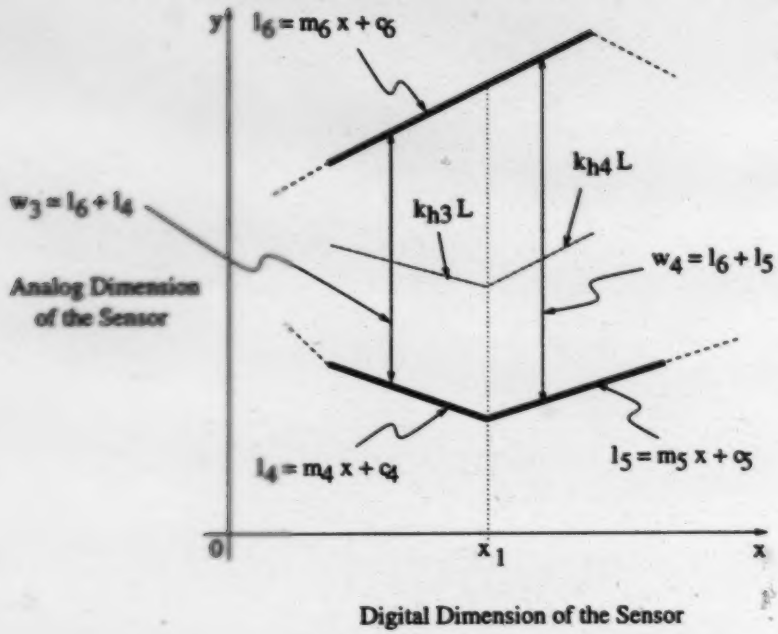


Figure 4.8: Signature Transition from One Section to Another with a Vertex at the Bottom

The gradients of the equations (4.8) to (4.11) are given by

$$\frac{d(k_{h1}L)}{dx} = \frac{L(m_1+m_2)}{2L - m_2x - c_2 + m_1x + c_1} - \frac{L(m_1x + c_1 + m_2x + c_2)(m_1 - m_2)}{(2L - m_2x - c_2 + m_1x + c_1)^2} \quad (4.12)$$

$$\frac{d(k_{h2}L)}{dx} = \frac{L(m_1+m_3)}{2L - m_3x - c_3 + m_1x + c_1} - \frac{L(m_1x + c_1 + m_3x + c_3)(m_1 - m_3)}{(2L - m_3x - c_3 + m_1x + c_1)^2} \quad (4.13)$$

$$\frac{d(k_{h3}L)}{dx} = \frac{L(m_4+m_6)}{2L - m_6x - c_6 + m_4x + c_4} - \frac{L(m_4x + c_4 + m_6x + c_6)(m_4 - m_6)}{(2L - m_6x - c_6 + m_4x + c_4)^2} \quad (4.14)$$

$$\frac{d(k_{h4}L)}{dx} = \frac{L(m_5+m_6)}{2L - m_6x - c_6 + m_5x + c_5} - \frac{L(m_5x + c_5 + m_6x + c_6)(m_5 - m_6)}{(2L - m_6x - c_6 + m_5x + c_5)^2} \quad (4.15)$$

Assume that x_0 and x_1 are the x-coordinates of the transitional points in Figure 4.7 and Figure 4.8 respectively. The best scenarios are $k_{h1}L$ and $k_{h2}L$ having their first-order derivatives discontinuous at $x = x_0$, or those of $k_{h3}L$ and $k_{h4}L$ being discontinuous at $x = x_1$. With these characteristic discontinuities, relatively simple gradient change detection techniques can be applied to obtained the transitional points from a signature. Unfortunately, there exist some combinations of m_1, m_2, m_3, c_1, c_2, c_3 and L that such discontinuities vanish. These combinations are

(a) $c2 = \frac{m_1L + m_2c_1 + m_2L}{m_1}$ and $c3 = \frac{m_1L + m_3L + m_3c_1}{m_1}$.
(b) $c4 = \frac{m_4c_6 - m_4L - m_6L}{m_6}$ and $c5 = \frac{m_5c_6 - m_5L - m_6L}{m_6}$.

2. In case of k_l, the signature segments are given by

$$k_{l1}L = \frac{l_1}{L - \frac{w_1}{2}} \times L = \frac{2L(m_1x + c_1)}{2L - m_2x - c_2 + m_1x + c_1} \tag{4.16}$$

$$k_{l2}L = \frac{l_1}{L - \frac{w_2}{2}} \times L = \frac{2L(m_1x + c_1)}{2L - m_3x - c_3 + m_1x + c_1} \tag{4.17}$$

$$k_{l3}L = \frac{l_4}{L - \frac{w_3}{2}} \times L = \frac{2L(m_4x + c_4)}{2L - m_6x - c_6 + m_4x + c_4} \tag{4.18}$$

$$k_{l4}L = \frac{l_5}{L - \frac{w_4}{2}} \times L = \frac{2L(m_5x + c_5)}{2L - m_6x - c_6 + m_5x + c_5} \tag{4.19}$$

The derivatives of the equations (4.16) to (4.19) are given by

$$\frac{d(k_{l1}L)}{dx} = \frac{2Lm_1}{2L - m_2x - c_2 + m_1x + c_1} - \frac{2L(m_1x + c_1)(m_1x - m_2)}{(2L - m_2x - c_2 + m_1x + c_1)^2} \tag{4.20}$$

$$\frac{d(k_{l2}L)}{dx} = \frac{2Lm_1}{2L - m_3x - c_3 + m_1x + c_1} - \frac{2L(m_1x + c_1)(m_1 - m_3)}{(2L - m_3x - c_3 + m_1x + c_1)^2} \tag{4.21}$$

$$\frac{d(k_{l3}L)}{dx} = \frac{2Lm_4}{2L - m_6x - c_6 + m_4x + c_4} - \frac{2L(m_4x + c_4)(m_4 - m_6)}{(2L - m_6x - c_6 + m_4x + c_4)^2} \tag{4.22}$$

$$\frac{d(k_{l4}L)}{dx} = \frac{2Lm_5}{2L - m_6x - c_6 + m_5x + c_5} - \frac{2L\,(m_5x + c_5)\,(m_5 - m_6)}{(2L - m_6x - c_6 + m_5x + c_5)^2} \tag{4.23}$$

Assume that x_0 and x_1 are the x-coordinates of the transitional points in Figure 4.7 and Figure 4.8 respectively. Similar to the case in k_h, the best scenarios are $k_{l1}L$ and $k_{l2}L$ having their first-order derivatives discontinuous at $x = x_0$, or those of $k_{l3}L$ and $k_{l4}L$ being discontinuous at $x = x_1$, which facilitate transitional points detection. However, such discontinuities do not always exist. The following combinations of m_1, m_2, m_3, c_1, c_2, c_3 and L defeat the discontinuities:

(a) $m_1 = 0$ and $c_1 = 0$ simultaneously. Fortunately, this condition is practically impossible since it implies that an entire edge of the polygon is touching the true zero reference.

(b) $c2 = \frac{m_1c_3 - m_3c_1 + m_2c_1}{m_1}$ and $c3 = \frac{2m_1L + m_3c_1}{m_1}$.

(c) $c4 = \frac{m_4(c_6 - 2L)}{m_6}$ and $c5 = \frac{m_5(c_6 - 2L)}{m_6}$.

3. In case of k, the signature segments are given by

$$k_1L = \frac{l_1}{L - w_1} \times L = \frac{L\,(m_1x + c_1)}{L - m_2x - c_2 + m_1x + c_1} \tag{4.24}$$

$$k_2L = \frac{l_1}{L - w_2} \times L = \frac{L\,(m_1x + c_1)}{L - m_3x - c_3 + m_1x + c_1} \tag{4.25}$$

$$k_3L = \frac{l_4}{L - w_3} \times L = \frac{L\,(m_4x + c_4)}{L - m_6x - c_6 + m_4x + c_4} \tag{4.26}$$

$$k_4L = \frac{l_5}{L - w_4} \times L = \frac{L\,(m_5x + c_5)}{L - m_6x - c_6 + m_5x + c_5} \tag{4.27}$$

The derivatives of the equations (4.24) to (4.27) are given by

$$\frac{d(k_1L)}{dx} = \frac{Lm_1}{L - m_2x - c_2 + m_1x + c_1} - \frac{L(m_1x + c_1)(m_1x - m_2)}{(L - m_2x - c_2 + m_1x + c_1)^2} \tag{4.28}$$

$$\frac{d(k_2L)}{dx} = \frac{Lm_1}{L - m_3x - c_3 + m_1x + c_1} - \frac{L(m_1x + c_1)(m_1 - m_3)}{(L - m_3x - c_3 + m_1x + c_1)^2} \tag{4.29}$$

$$\frac{d(k_3L)}{dx} = \frac{Lm_4}{L - m_6x - c_6 + m_4x + c_4} - \frac{L(m_4x + c_4)(m_4 - m_6)}{(L - m_6x - c_6 + m_4x + c_4)^2} \tag{4.30}$$

$$\frac{d(k_4L)}{dx} = \frac{Lm_5}{L - m_6x - c_6 + m_5x + c_5} - \frac{L(m_5x + c_5)(m_5 - m_6)}{(L - m_6x - c_6 + m_5x + c_5)^2} \tag{4.31}$$

Assume that x_0 and x_1 are the x-coordinates of the transitional points in Figure 4.7 and Figure 4.8 respectively. Similar to the cases of k_h and k_l, the best scenarios are k_1L and k_2L having their first-order derivatives discontinuous at $x = x_0$, or those of k_3L and k_4L being discontinuous at $x = x_1$, which facilitate transitional points detection. However, such discontinuities vanish with the following combinations of m_1, m_2, m_3, c_1, c_2, c_3 and L:

(a) $m_1 = 0$ and $c_1 = 0$ simultaneously. Fortunately, this condition is practically impossible as explained in the case of k_l.

(b) $c2 = \frac{m_1c_3 - m_3c_1 + m_2c_1}{m_1}$ and $c3 = \frac{m_1L + m_3c_1}{m_1}$.

(c) $m_6 = 0$ and $c_6 = L$ simultaneously. This condition is practically impossible since it implies that an entire edge of a polygon is touching the upper edge of the sensor.

(d) $c4 = \frac{m_4(c_6 - L)}{m_6}$ and $c5 = \frac{m_5(c_6 - L)}{m_6}$.

In conclusion, no matter what kind of pseudo-signature is used, there are only a few locations and orientations of a polygonal contact shape that produce pseudo-signature whose transitional points are difficult to detect.

4.2.4 Recovering the Vertices

The Vertex Reconstruction algorithm is based on the linearity of the edges of the polygonal shape. Figure 4.9(a) to (c) illustrate how a system of linear equations can be set up to recover the interior vertices. Though an arbitrary hexagon and its k_h-pseudo-signature is used in the illustration, the basic principle is applicable to any polygonal shape and pseudo-signature discussed before. As shown in Figure 4.9(a), the highlighted portion of

the polygon enables the formulation of the following equations:

$$\begin{cases} k_{h2} = \dfrac{y_{t2} + \dfrac{y_2 - y_{t2}}{2}}{L - \dfrac{y_2 - y_{t2}}{2}} \\ \dfrac{y_{t2} - y_1}{x_2 - x_1} = \dfrac{y_3 - y_1}{x_3 - x_1} \end{cases}$$

$$\Rightarrow \quad \frac{1 + k_{h2}}{1 - k_{h2}} y_2 + \frac{x_2 - x_1}{x_3 - x_1} y_3 = \frac{2k_{h2}L}{1 - k_{h2}} - \frac{x_3 - x_2}{x_3 - x_1} y_1 \tag{4.32}$$

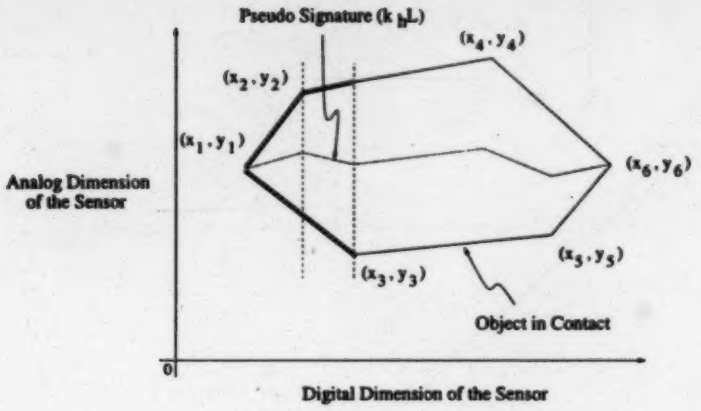
For the next portion of the shape as shown in Figure 4.9(b), another set of linear equations is formed by the current pairs of interior vertices:

$$\begin{cases} k_{h3} = \dfrac{y_3 + \dfrac{y_{h3} - y_3}{2}}{L - \dfrac{y_{h3} - y_3}{2}} \\ \dfrac{y_{h3} - y_2}{x_3 - x_2} = \dfrac{y_4 - y_2}{x_4 - x_2} \end{cases}$$

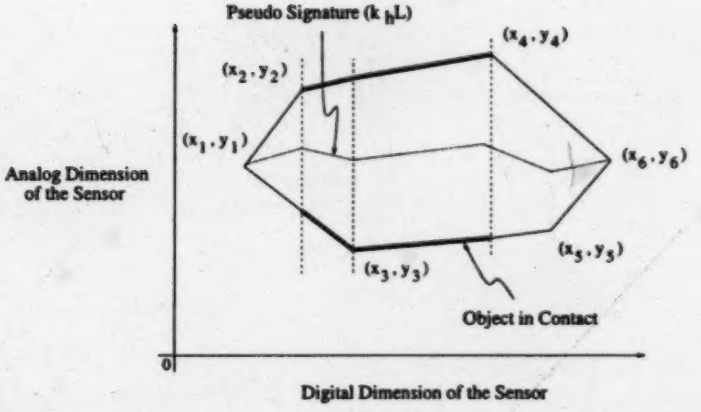
$$\Rightarrow \quad \frac{x_4 - x_3}{x_4 - x_2} y_2 + \frac{1 - k_{h3}}{1 + k_{h3}} y_3 + \frac{x_3 - x_2}{x_4 - x_2} y_4 = \frac{2k_{h3}L}{1 + k_{h3}} \tag{4.33}$$

Similar process continues until all portions of the shape are considered (see Figure 4.9(c)). As a result, the corresponding system of linear equations is obtained as follows:

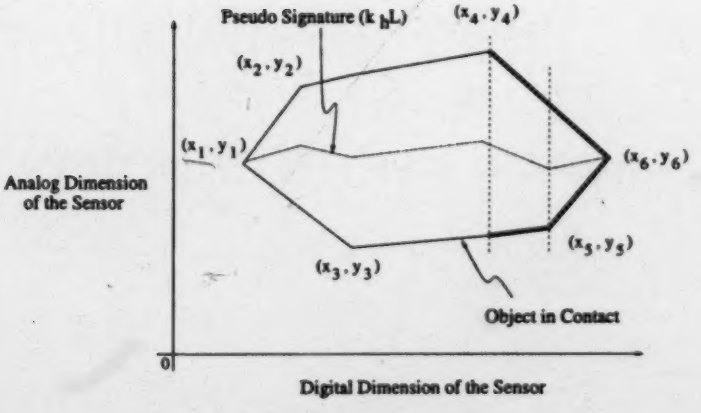
$$\begin{bmatrix} \frac{1 + k_{h2}}{1 - k_{h2}} & \frac{x_2 - x_1}{x_3 - x_1} & 0 & 0 \\ \frac{x_4 - x_3}{x_4 - x_2} & \frac{1 - k_{h3}}{1 + k_{h3}} & \frac{x_3 - x_2}{x_4 - x_2} & 0 \\ 0 & \frac{x_5 - x_4}{x_5 - x_3} & \frac{1 + k_{h4}}{1 - k_{h4}} & \frac{x_4 - x_3}{x_5 - x_3} \\ 0 & 0 & \frac{x_6 - x_5}{x_6 - x_4} & \frac{1 - k_{h5}}{1 + k_{h5}} \end{bmatrix} \begin{bmatrix} y_2 \\ y_3 \\ y_4 \\ y_5 \end{bmatrix} = \begin{bmatrix} \frac{2k_{h2}L}{1 - k_{h2}} - \frac{x_3 - x_2}{x_3 - x_1} y_1 \\ \frac{2k_{h3}L}{1 + k_{h3}} \\ \frac{2k_{h4}L}{1 - k_{h4}} \\ \frac{2k_{h5}L}{1 + k_{h5}} - \frac{x_5 - x_4}{x_6 - x_4} y_6 \end{bmatrix} \tag{4.34}$$



(a)



(b)



(c)

Figure 4.9: Geometric Signature and Pseudo-Signatures of an Arbitrary Tetragon

As mentioned in Section 4.2.2, the two end vertices of a polygon shape are readily found by setting $w = 0$ in corresponding equation of either k_h, k_l or k, and a system of linear equations is required to solve for the interior vertices of the polygonal shape. In general, a N-sided polygon has a system of $N-2$ linear equations in the following form:

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1(N-2)} \\ a_{21} & a_{22} & \cdots & a_{2(N-2)} \\ \cdots & \cdots & \cdots & \cdots \\ a_{(N-2)1} & a_{(N-2)2} & \cdots & a_{(N-2)(N-2)} \end{bmatrix} \begin{bmatrix} y_2 \\ y_3 \\ \cdots \\ y_{N-1} \end{bmatrix} = \begin{bmatrix} b_1 \\ b_2 \\ \cdots \\ b_{N-2} \end{bmatrix} \tag{4.35}$$

The Vertex Reconstruction algorithm is used to set up the coefficients of the above matrix A (i.e. a_{11}, a_{12}, ..., $a_{(N-2)(N-2)}$) and matrix B (i.e. b_1, b_2, ..., b_{N-2}) with the following steps:

1. The algorithm starts by identifying any signature on the sensor. Consecutive non-zero values of k_h, k_l or k constitute a signature. As discussed in previous sections, the signature is a set of piece-wise continuous line segments with transitional points corresponding to vertices of the contact shape. The polygonal shape is virtually chopped into $N-1$ vertical sections.

2. The orientation of the signature segment in a section is approximated by a linear line segment joining its two end-points. The algorithm works from the left-most vertical section and takes it as the current section. It compares the orientation of the signature in the current section to that in the next section to the right. If they are both clockwise (or anti-clockwise), this next section will be the new current section and the comparison goes on. It keeps track of the number of consecutive sections with the same signature orientation until a different orientation occurs in the next section. Such comparison continues until the right-most vertical section is checked. Figure 4.10 illustrates how the orientations of the pseudo-signature segments are compared. The consecutive sections with the same signature orientations are called blocks.

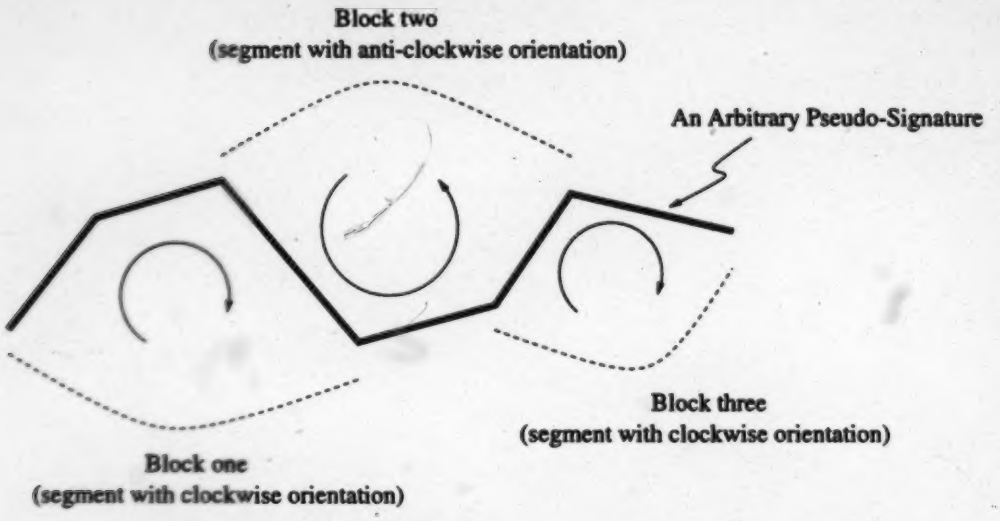


Figure 4.10: An Arbitrary Pseudo-Signature with Three Blocks of Sections

3. Initialize all coefficients in matrix A to zeros. Then assign the diagonal coefficients of matrix A and initialize matrix B as below:

   (a) In case of k_h (i.e. k_{h1}, k_{h2}, ..., k_{hN} obtained at the sensing elements corresponding to the transitional points):

   for $i = 2$ to $N - 1$

   if vertex i is above the transition

$$a_{(i-1)(i-1)} = \frac{1 + k_{hi}}{1 - k_{hi}}$$

$$b_{i-1} = \frac{2k_{hi}L}{1 - k_{hi}}$$

   else

$$a_{(i-1)(i-1)} = \frac{1 - k_{hi}}{1 + k_{hi}}$$

$$b_{i-1} = \frac{2k_{hi}L}{1 + k_{hi}}$$

(b) In case of k_l,

for $i = 2$ to $N - 1$

if vertex i is above the transition

$$a_{(i-1)(i-1)} = \frac{k_{li}}{2 - k_{li}}$$

$$b_{i-1} = \frac{2k_{li}L}{2 - k_{li}}$$

else

$$a_{(i-1)(i-1)} = \frac{2 - k_{li}}{k_{li}}$$

$$b_{i-1} = 2L$$

(c) In case of k,

for $i = 2$ to $N - 1$

if vertex i is above the transition

$$a_{(i-1)(i-1)} = \frac{k_i}{1 - k_i}$$

$$b_{i-1} = \frac{k_i L}{1 - k_i}$$

else

$$a_{(i-1)(i-1)} = \frac{1 - k_i}{k_i}$$

$$b_{i-1} = L$$

4. For the blocks obtained in step 2, there are four possible combinations of their corresponding vertex types (see Table 4.1). The coefficients of matrices A and B are then assigned according to the following rules with respect to these combinations:

Legend: *start* is the index to the first vertex of a block

end is the index to the last vertex of a block

(a) If a block is of type (I) in Table 4.1,

for $row = start$ to $end - 2$

$$a_{(row)(end-1)} = \frac{x_{row+1} - x_{start}}{x_{end} - x_{start}}$$

$$b_{row} = b_{row} - \frac{x_{end} - x_{row+1}}{x_{end} - x_{start}} y_{start}$$

(b) If a block is of type (II) in Table 4.1,

for $row = start$ to $end - 2$

$$b_{row} = b_{row} - \frac{x_{end} - x_{row+1}}{x_{end} - x_{start}} y_{start} - \frac{x_{row+1} - x_{start}}{x_{end} - x_{start}} y_{end}$$

(c) If a block is of type (III) in Table 4.1,

for $row = start$ to $end - 2$

$$a_{(row)(start-1)} = \frac{x_{end} - x_{row+1}}{x_{end} - x_{start}}$$
$$b_{row} = b_{row} - \frac{x_{row+1} - x_{start}}{x_{end} - x_{start}} y_{end}$$

(d) If a block is of type (IV) in Table 4.1,

for $row = start$ to $end - 2$

$$a_{(row)(start-1)} = \frac{x_{end} - x_{row+1}}{x_{end} - x_{start}}$$
$$a_{(row)(end-1)} = \frac{x_{row+1} - x_{start}}{x_{end} - x_{start}}$$

Block Type	Start Vertex	End Vertex	Action on
I	Left-most Vertex	Any Interior Vertex	Both Matrix A & B
II	Left-most Vertex	Right-most Vertex	Matrix B Only
III	Any Interior Vertex	Right-most Vertex	Both Matrix A & B
IV	Any Interior Vertex	Any Interior Vertex	Matrix A Only

Table 4.1: Block Types

4.3 Profile Reconstruction

Profile Reconstruction is applied if double-measurement is available. It takes both the arrays of k_l and k_h and generates the profile of the contact shape. There are also some assumptions and limitations that need to be addressed before applying this algorithm.

4.3.1 Assumptions on Shapes

In general, there is no need to assume that the contact shape is polygonal and limited concavity is allowed. However, the shape must not have concavity of the form illustrated in Figure 4.5(b).

4.3.2 Recovering the Profile

With both sets of k_l and k_h, it is possible to recover both values of l and w of each column of the sensor. By solving the equations 3.3 and 3.4 simultaneously, the values of the corresponding l and w are readily found as follows:

$$\begin{bmatrix} 1 & \frac{1+k_h}{2} \\ 1 & \frac{k_l}{2} \end{bmatrix} \begin{bmatrix} l \\ w \end{bmatrix} = \begin{bmatrix} k_h L \\ k_l L \end{bmatrix} \tag{4.36}$$

Hence, l and w are solved for each sensor column as below:

$$l = \frac{k_l L}{1 + k_h - k_l}$$

$$w = \frac{2(k_h - k_l)L}{1 + k_h - k_l}$$

In general, the profile of an object can be obtained by solving all l and w for each sensor column in contact with the object. It is noticed that both l and w of a column can be solved independent of other columns. There is no reference to the trace of the pseudo-centers. Therefore, the ambiguity due to concavity shown in Figure 4.5(a) is not a concern in this algorithm.

	Vertex Reconstruction	Profile Reconstruction
Applicable Sensor	CRS-CRS & CRS-M	CRS-CRS
Transition Point Detection	Required	Not Required
Limitation on Shapes	Convex Polygons Only	Any Convex Shapes & Certain Concave Shapes
Data Size (With n Sensing Elements)	n	$2n$

Table 4.2: Comparison Between Vertex Reconstruction and Profile Reconstruction

4.3.3 Comparison Between Vertex Reconstruction and Profile Reconstruction

Table 4.2 gives a comparison between Vertex Reconstruction and Profile Reconstruction. Vertex Reconstruction requires only k_h, k_l or k from the sensor. It can be used with both the CRS-CRS and CRS-M sensors. However, it is only capable of recovering convex polygonal

shapes and transition points detection must be done on the raw data (pseudo-signature) to determine the x-coordinates of the vertices being recovered. Profile Reconstruction, on the other hand, makes use of both k_h and k_l. Hence, it requires twice as many data as that in Vertex Reconstruction. Since only CRS-CRS sensor provides both k_h and k_l, Profile Reconstruction cannot be used with a CRS-M sensor. As Profile Reconstruction estimates l and w for each sensing element, it is capable of recovering any convex shapes and certain concave shapes.

Chapter 5

Simulations

To verify the shape recovery algorithms with different types of hybrid sensors, a number of simulations have been done using Matlab. Various parameters, such as errors in sensor outputs, geometric shape of the object in contact, its position and orientation relative to the sensor were used.

5.1 Sources of Errors

In practice, a number of sources contribute to errors in the sensor output, which in turns affect the performance of a shape recovery algorithm. To facilitate simulations and analysis, the possible error sources are grouped into two major categories, namely, quantization noise and positional uncertainty.

5.1.1 Combined Quantization Noise

As mentioned in previous Chapters, an analog-to-digital (A/D) conversion is required to turn the analog voltage into a digital form for computer processing. The maximum uncertainty in this conversion is half the resolution used. If the output voltage is digitized into a 12-bit number, the A/D quantization error will be $\pm 1/8192$ of the full scale output voltage. Besides this A/D quantization error, another kind of quantization takes place on the length of contact between an object and the sensor. Figure 5.1 illustrates how the edges of a polygon are quantized. Since each sensor column has a finite width, the measured output voltage corresponds to the average position of contact. Each of the upper and lower contact ends can have a maximum error of half the resolution. If the analog dimension is geometrically divided into 1000 units, maximum error at each contact end will be $\pm 1/2000$. Hence in the worst case, the maximum quantization error in a length of contact adds up to $\pm 1/1000$. As both of the above quantization errors are inherent in any hybrid sensor

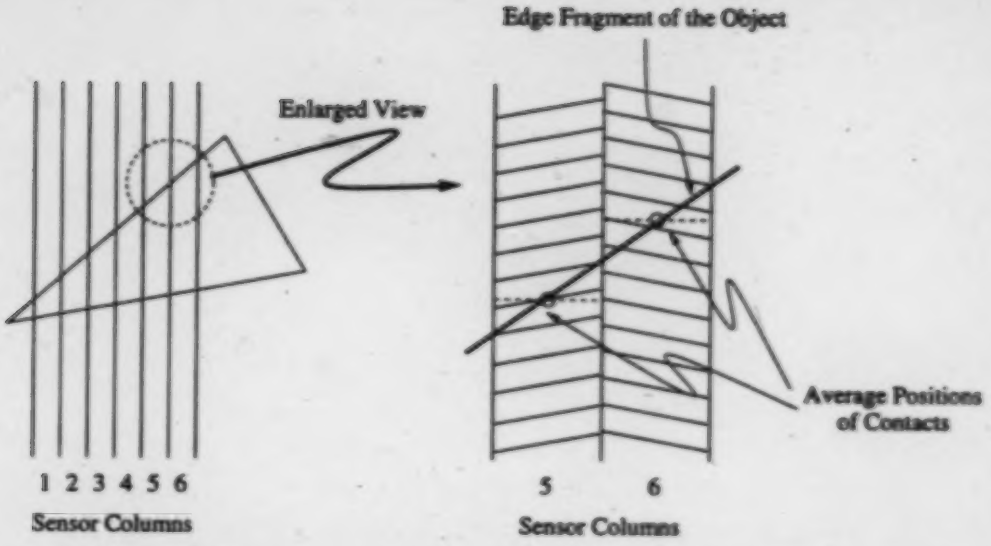


Figure 5.1: Quantization at the Edge of Contact between an Object and the Hybrid Sensor.

discussed in this thesis, they are grouped together and defined as *Combined Quantization Errors* (C.Q.E.).

5.1.2 Error in Detecting Transitions of a Pseudo-signature

As discussed in Chapter 4, a pseudo-signature is required for the Vertex Reconstruction algorithm to work. The detection of any transition in such a signature is particularly important. Even with ideal sensor output, tolerance needs to be allowed in any transition detecting algorithm. To maintain simplicity in simulations, a uniformly distributed uncertainty with a range of ± 5 units was randomly added to the horizontal coordinates of each transitions. This error is defined as *Transition Error* (T.E.).

5.1.3 Positional Uncertainty

Ideally, the contact between an object and the sensor is assumed to be perfect. However, in practical situations, it is unlikely to have perfect contact all the time. Figure 5.2 gives a typical example of imperfect contacts when an object touches a sensor surface. Factors such as the non-uniformity in CRS strips and slight mis-alignment between two CRS strips can lead to errors in the sensor output. These errors were categorised as *Positional Uncertainty* (P.U.). As there may be many combinations of these error sources, it is impossible to model the positional uncertainties precisely. In practice, P.U. are reflected as error in the measured output voltage. For instance, if δV_h is the error in the measured voltage V_h, then

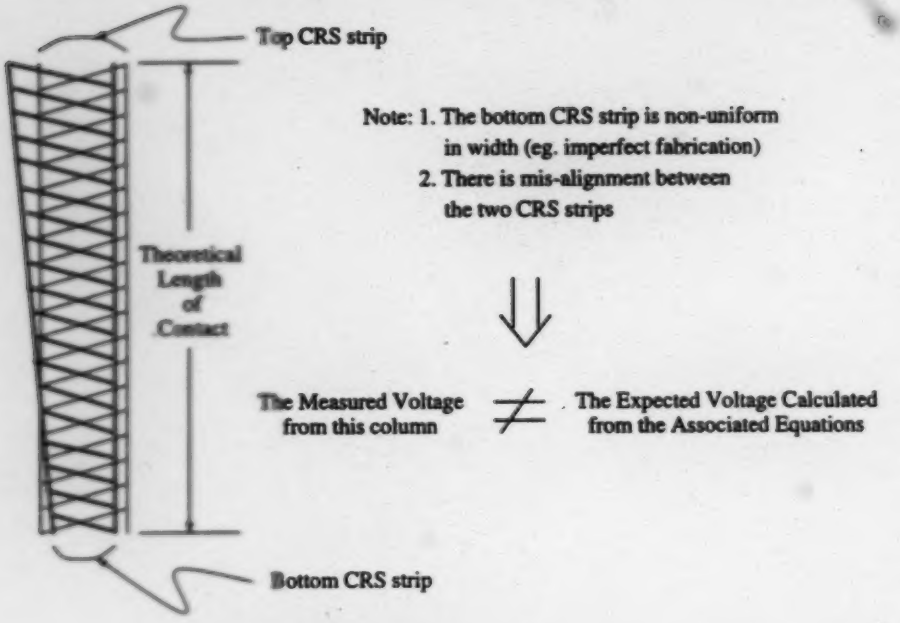


Figure 5.2: Typical Examples that Constitute Positional Uncertainty

the corresponding value of k_h will be given by:

$$\hat{k}_h = \frac{V_h + \delta V_h}{V_{ref}} = k_h + \frac{\delta V_h}{V_{ref}} = k_h + \delta k_h \tag{5.1}$$

In case of k_l and k, their values are given by:

$$\hat{k}_l = k_l + \frac{\delta V_l}{V_{ref}} = k_l + \delta k_l \tag{5.2}$$

$$\hat{k} = k + \frac{\delta V_{out}}{V_{ref}} = k + \delta k \tag{5.3}$$

where δV_l and δV_{out} are the errors in measured voltage V_l and V_{out} respectively.

As observed from equations 5.1 to 5.3, the errors due to P.U. can be represented by a percentage with respect to the reference voltage V_{ref}.

5.2 Simulated Scenarios

The simulations focus on the CRS-CRS and the CRS-M sensor structures. In the case of a CRS-CRS sensor, both the Vertex Reconstruction and the Profile Reconstruction algorithms

can be applied for shape recovery. However, only Vertex Reconstruction can be used with a CRS-M sensor. Three types of polygons, namely a triangle, a tetragon and a hexagon were used in the simulations. In each combination of the above sensor structures, shape recovery algorithms and polygonal shapes, five scenarios of shape recovery with different sensor output noise levels were simulated. These noise levels were:

1. Ideal Sensor Output (Noise-free Sensor).
2. Sensor Output with C.Q.E. (and T.E., if applicable) Only.
3. Sensor Output with C.Q.E. (and T.E., if applicable) and P.U. profile I.
4. Sensor Output with C.Q.E. (and T.E., if applicable) and P.U. profile II.

The scenarios of ideal sensors serve as control cases to verify the equations and their solutions as discussed in Chapter 4. In case of the Vertices Reconstruction Algorithm with any sensor structure, quantization noise was simulated in both the digital and the analog dimensions. The noise in the analog dimension was assumed to be the result of any analog-to-digital conversion, whereas the noise from the digital dimension was assumed to be the uncertainty in determining the horizontal coordinates of the transition of a pseudo-signature. On top of these quantization errors, two profiles (see Table 5.1) of positional uncertainties in the analog dimension were added in turns.

Profile	Percentage Error of Sensor Output With Respect To V_{ref}
I	± 0 to 0.5%
II	± 0 to 1%

Table 5.1: Simulated Error Profiles for Positional Uncertainty

In each simulated case, the hybrid sensor assumed a planar square geometry and its size was 1000 units in both the digital (horizontal) and the analog (vertical) dimensions. The simulated shapes were dragged across the sensor from the lower left corner to the upper right corner in 5 steps. They were also rotated by 36° counter-clockwise at each step along their trajectories. To facilitate discussion on the results, a number of metrics are required to measure how well the recovered shape matches the original.

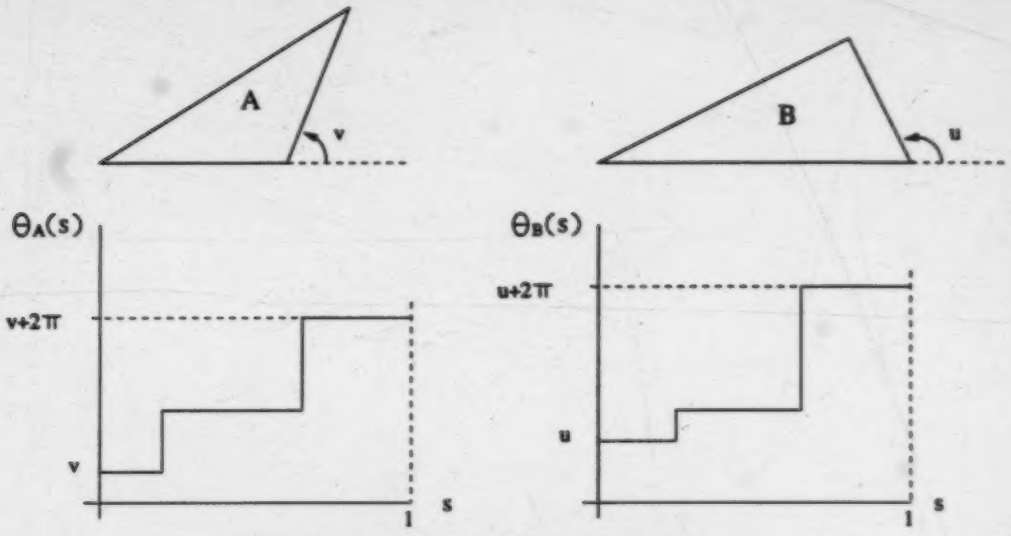


Figure 5.3: Turn Functions of Polygons A and B.

5.3 Metrics to Measure Shape Recovery Performance

This thesis focuses on the following characteristics when measuring how well the recovered shape matches the original:

(a) Shape: The similarity in profile between the original and the recovered shape.

(b) Position: The distance between the geometric centroids of the original and the recovered shape.

(c) Orientation: The deviation in orientation between the the original and the recovered shape.

In comparing similarity (or difference) in profile, the *Polygon Distance Function*, which is commonly used by researchers in computational geometry [17], has been adopted. Each polygonal shape is represented by its normalized turn function $\theta(s)$. Figure 5.3 shows the corresponding normalized turn functions for polygons A and B. The quantity, Mismatch Coefficient (M.C.), is defined as the sum of absolute difference between $\theta_A(s)$ and $\theta_B(s)$, as illustrated in Figure 5.4. In general, a perfect match corresponds to a 0 radians in M.C. and larger dis-similarity corresponds to higher values in M.C.

To measure how close the position of the recovered polygon to the original, the geometric centroids of the polygons are used as reference. The quantity, Absolute Position Error (A.P.E.), is defined as the distance between the geometric centroids of two polygons. The

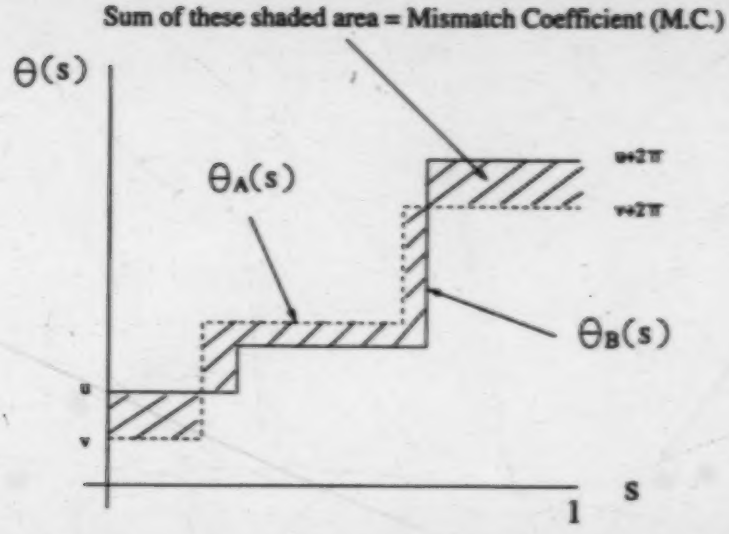


Figure 5.4: Mismatch Coefficient (M.C.) Between Polygons A and B shown in Figure 5.3.

two polygons are considered perfectly at the same position if the A.P.E. between them is 0. On the other hand, increasing A.P.E. means increasing discrepancy in positions between the two polygons.

The orientation of a polygon is defined as the angle between its reference line to the horizontal axis. The reference line is a line joining an arbitrary vertex to the geometric centroid. When comparing the orientations of the original and recovered polygons, their corresponding reference lines must be used. The quantity, Absolute Orientation Error (A.O.E.), is then defined as the difference between the orientations of the two polygons concerned. A perfect match results in a zero A.O.E., whereas increasing A.O.E. means larger deviation.

5.4 Simulation Results with a CRS-CRS Hybrid Sensor

With a CRS-CRS hybrid sensor, there is a choice of using either the Vertex Reconstruction or the Profile Reconstruction algorithm for shape recovery. The next two sections cover the simulation results of using these algorithms.

5.4.1 Vertex Reconstruction with a CRS-CRS Hybrid Sensor using k_A-Pseudo-Signature

The first set of simulations was done with a triangle on a CRS-CRS hybrid sensor using Vertex Reconstruction. In the first scenario, ideal conditions were assumed. With perfect

sensor output and transition detection in the pseudo-signatures, the values of k_h were used to recover the original shape perfectly (see Figure 5.5). In the second scenario, only C.Q.E. was introduced. The maximum quantization error was $\pm 1/1000$ (the geometric resolution in the analog dimension) in each of the concerned columns. These quantization noises were directly reflected in the sensor output. In addition, T.E. was assumed to be ranged from -5 to $+5$ units with a uniform distribution. Figure 5.6 shows the corresponding shape recovery. Graphically, the recovered shapes almost coincide with the original. In the next two scenarios, P.U. profiles I and II were added on top of the C.Q.E. and T.E. to the sensor output. Their simulation results are depicted in Figure 5.7 and Figure 5.8 respectively. Table 5.2 shows the corresponding quantitative errors in each scenario. The performance was satisfactory with P.U. profile I and the accuracy in shape recovery deteriorated with P.U. profile II. As shown in Figure 5.8, the algorithm failed to recover shape 5 and larger discrepancy was found in recovering the other shapes, especially in shape 1. This phenomenon was contributed by the following reasons:

1. Near the lower region, the relative error in k_h is large for given profile of P.U. For example, the dynamic range of the noise-free k_h is usually small for a typical polygonal shape, from 0.05 to 0.12 in shape 1 for instance. If P.U. causes a 1% error in V_h, the corresponding error in k_h will be 10%. As the recovery algorithm based heavily on the accuracy of k_h, larger error in k_h causes larger discrepancies between the recovered shapes and their originals.

2. Near the upper region, a given P.U. causes a smaller relative error in k_h. For example, the values of k_h is relatively large (from 0.86 to 0.87 in shape 5). If P.U. causes a 1% error in V_h, the corresponding error in k_h will be only around 1%. However the dynamic range of k_h in this region is relatively narrow, the effect of 1% error in k_h can cause large distortion in the signature. Figure 5.9 shows the noisy signature of shape 5 compared to its noise-free counterpart. By using the simple detection method in the simulation, too many erroneous transitional points were detected from this noisy signature and shape recovery was failed.

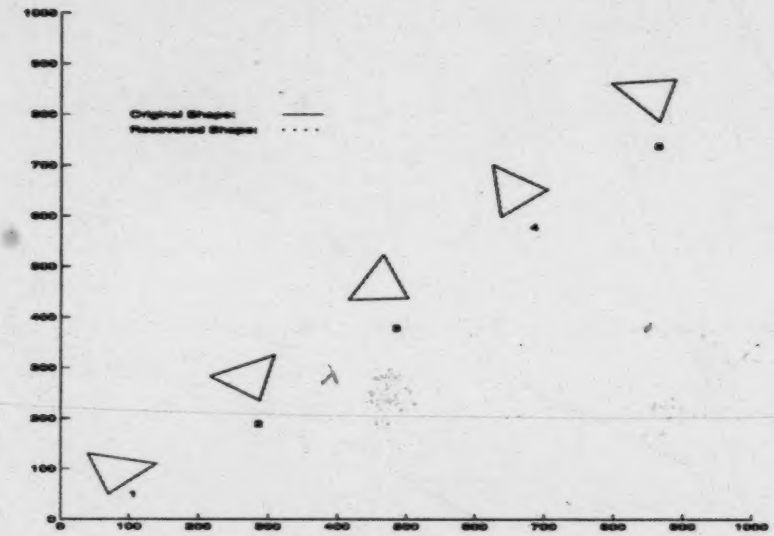


Figure 5.5: Recovering a Triangle on an ideal CRS-CRS Sensor by V.R. with k_h-Pseudo-Signature

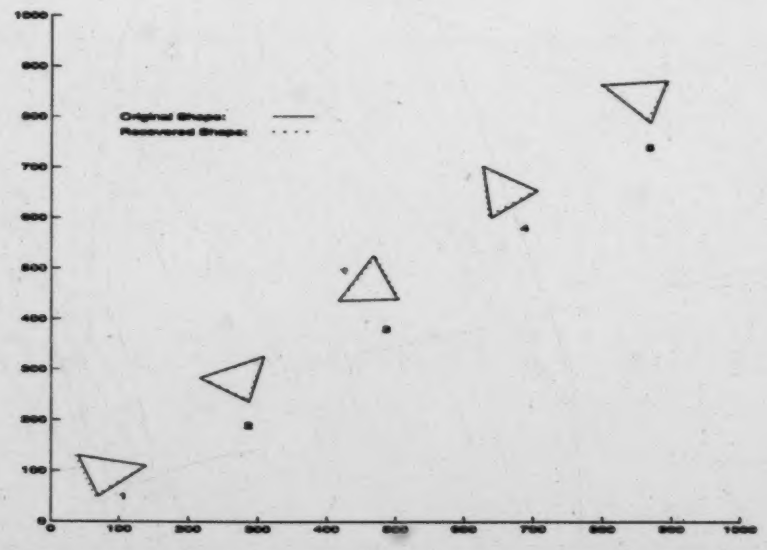


Figure 5.6: Recovering a Triangle on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_h-Pseudo-Signature

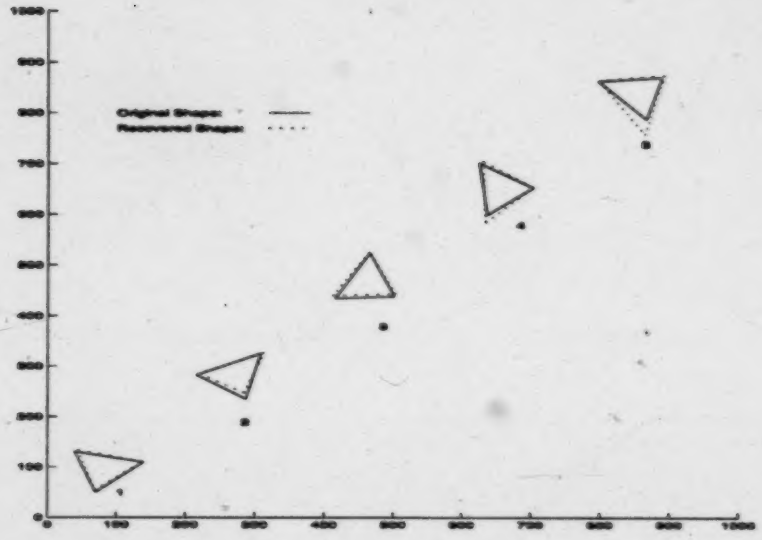


Figure 5.7: Recovering a Triangle on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_h-Pseudo-Signature

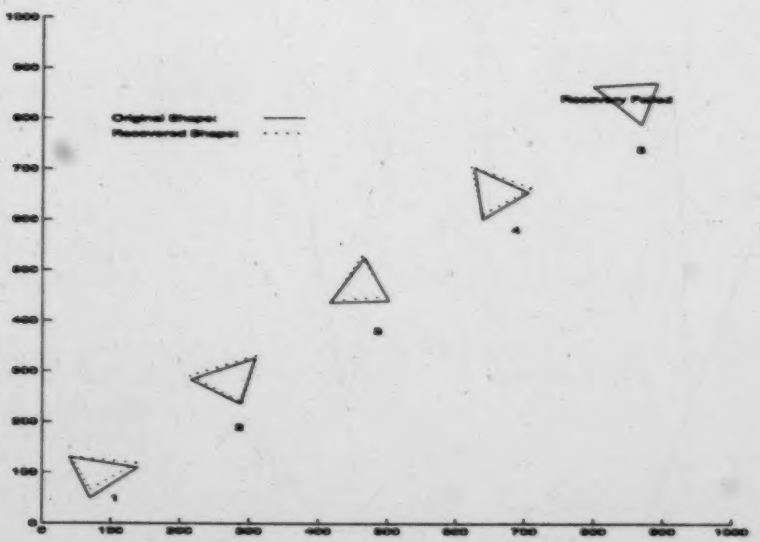


Figure 5.8: Recovering a Triangle on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. P.U. Profile II) by V.R. with k_h-Pseudo-Signature

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Vertex Reconstruction Signature Type: k_h Simulated Shape: Triangle						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.5	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.6	M.C.	0.031	0.048	0.020	0.039	0.059
	A.O.E.	0.007	0.019	0.023	0.001	0.049
	A.P.E.	3.670	1.167	2.333	2.580	2.775
Figure 5.7	M.C.	0.060	0.068	0.072	0.164	0.203
	A.O.E.	0.010	0.018	0.035	0.076	0.080
	A.P.E.	3.064	3.771	3.156	8.340	14.122
Figure 5.8	M.C.	0.194	0.077	0.080	0.135	-
	A.O.E.	0.015	0.029	0.025	0.080	-
	A.P.E.	6.544	3.153	3.614	7.653	-

Table 5.2: Quantitative Mismatch between the Original and the Recovered Triangle in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_h-Pseudo-Signature).

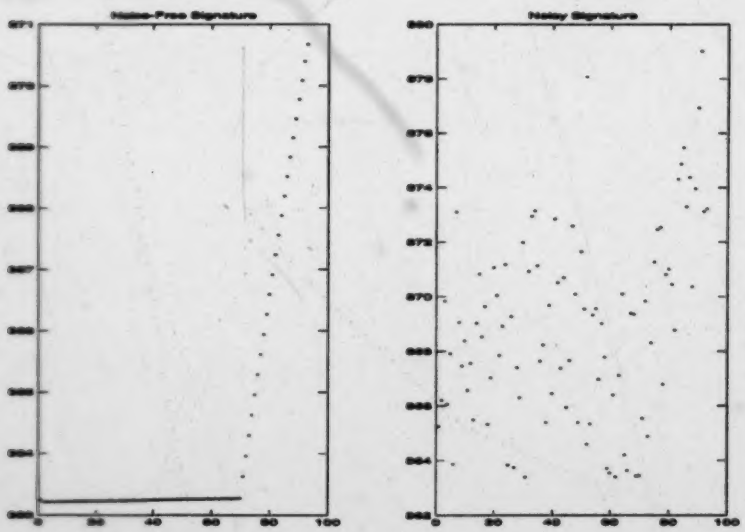


Figure 5.9: Noise-free and Noisy k_h-Pseudo-Signature of Triangular Shape 5 with P.U. Profile II

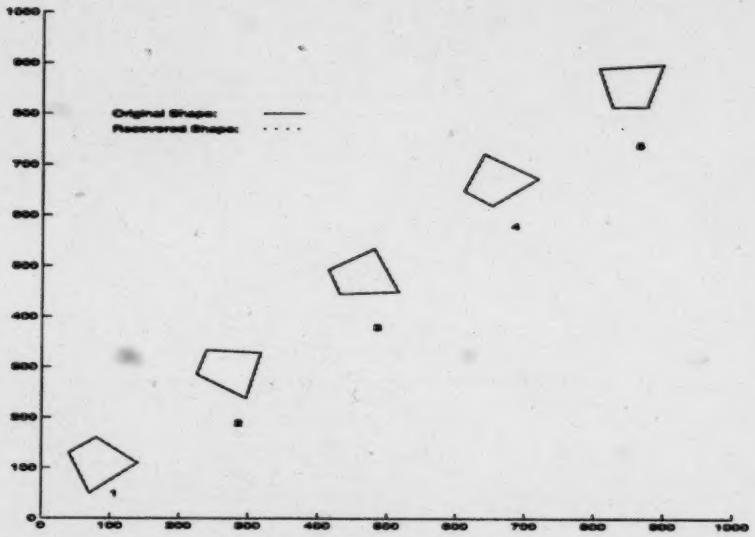


Figure 5.10: Recovering a Tetragon on an ideal CRS-CRS Sensor by V.R. with k_h-Pseudo-Signature

The second set of simulations were done with a tetragon. Figure 5.10 to Figure 5.13 show the corresponding simulation results and Table 5.3 gives their quantitative errors. With ideal sensor output, Vertex Reconstruction recovered the original tetragon perfectly. When only C.Q.E. and T.E. were added, only minor errors were found in shape recovery. Even with P.U. profile I, the tetragons could still be reconstructed. When P.U. profile II was used, the algorithm failed in recovering the shapes 1 and 5 as their signature were too noisy for transitional points detection. As seen in Figure 5.13, even shape recovery was possible, large discrepancy exist in recovering shape 4. The signature was so seriously distorted that one of the transitional points was not detected (see Figure 5.14). As a result, one interior vertex was missing and the recovered shape was more like a triangle rather than a tetragon.

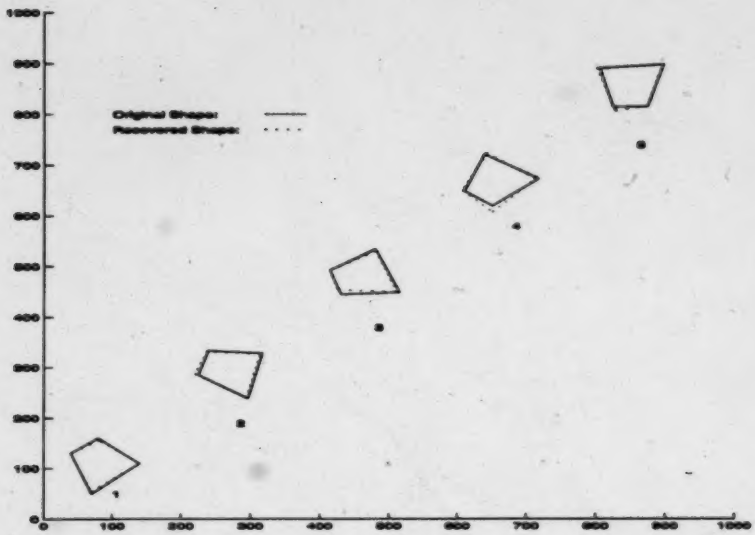


Figure 5.11: Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_h-Pseudo-Signature

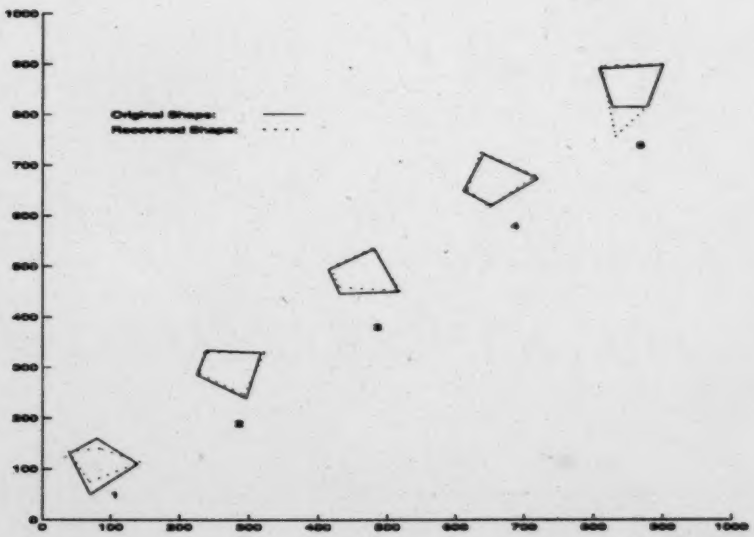


Figure 5.12: Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_h-Pseudo-Signature

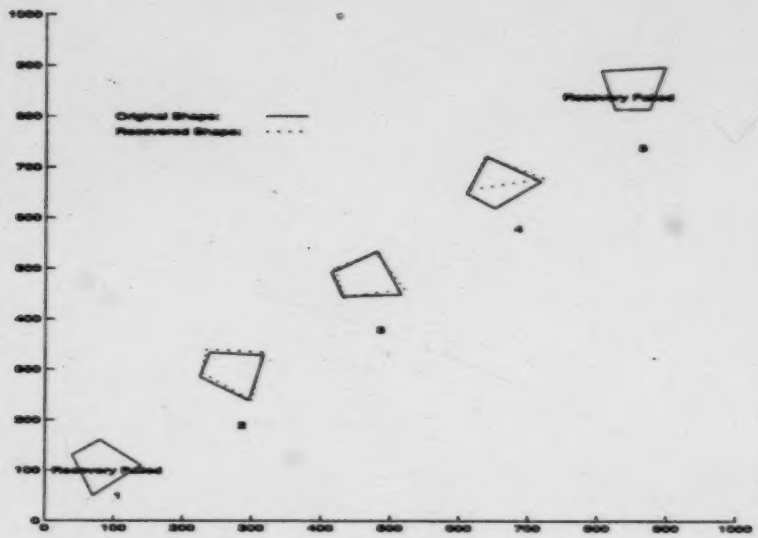


Figure 5.13: Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_h-Pseudo-Signature

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Vertex Reconstruction Signature Type: k_h Simulated Shape: Tetragon						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.10	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.11	M.C.	0.059	0.056	0.172	0.267	0.290
	A.O.E.	0.011	0.038	0.020	0.089	0.011
	A.P.E.	2.549	3.018	2.321	2.371	1.881
Figure 5.12	M.C.	0.326	0.105	0.128	0.075	0.413
	A.O.E.	0.082	0.025	0.018	0.015	0.132
	A.P.E.	6.517	1.482	2.335	2.214	19.679
Figure 5.13	M.C.	-	0.197	0.211	-	-
	A.O.E.	-	0.035	0.085	-	-
	A.P.E.	-	3.745	3.822	-	-

Table 5.3: Quantitative Mismatch between the Original and the Recovered Tetragon in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_h-Pseudo-Signature).

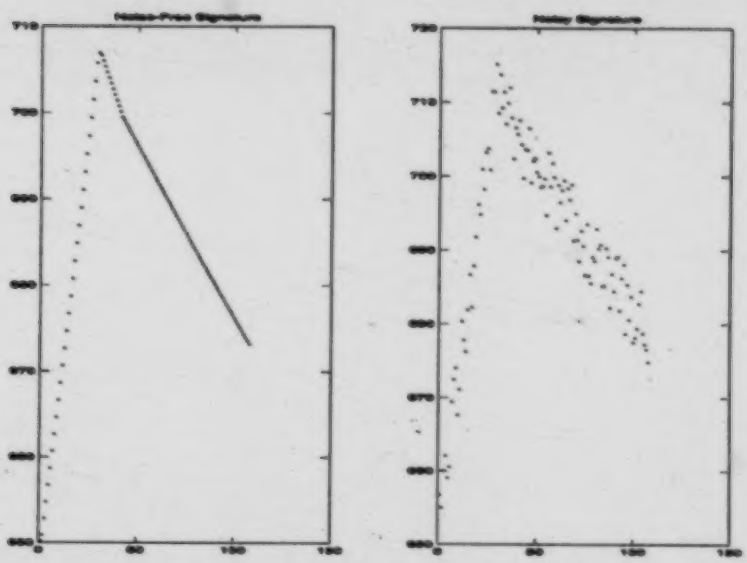


Figure 5.14: Noise-free and Noisy k_h-Pseudo-Signature of Tetragonal Shape 4 with P.U. Profile II

The third set of simulations was done with a hexagon. Figure 5.15 to Figure 5.18 show their corresponding simulation results under various error conditions. Table 5.4 summarizes their quantitative errors in each case. Similar to the results using a triangle and a tetragon, the algorithm reconstructed the original hexagon perfectly when an ideal sensor was simulated. Errors in shape recovery were still acceptable when C.Q.E. and T.E., and P.U. profile I were assumed. However, the algorithm started to perform poorly when P.U. profile II was used (see Figure 5.18). Since the dynamic range of k_h for a particular shape is usually small (from 0.1 to 0.2), it is difficult to extract transitional point from a seriously distorted signature. As the number of vertices increases, the chance of missing transitional points from the signature increases.

As mentioned in Section 4.2.1, Vertex Reconstruction algorithm is only applicable on convex polygon. Figure 5.19 shows the possible ambiguity it caused when the algorithm is used on a concave shape.

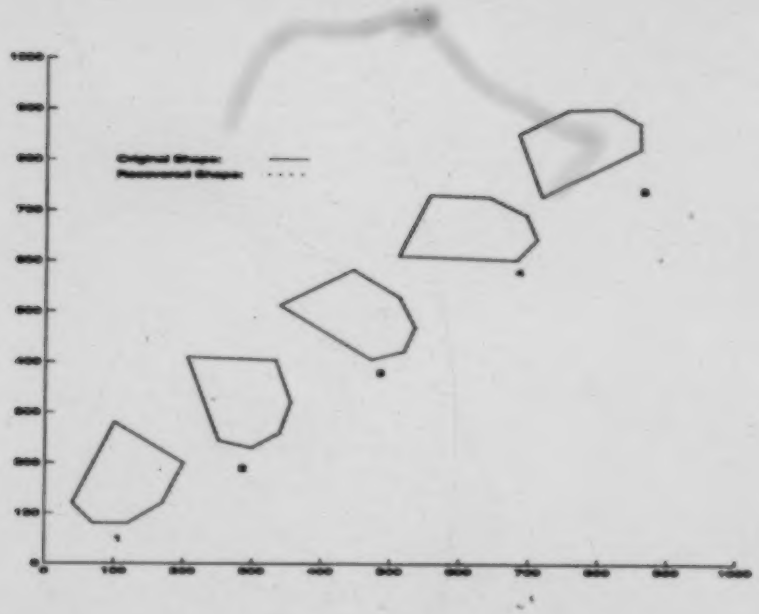


Figure 5.15: Recovering a Hexagon on an ideal CRS-CRS Sensor by V.R. with k_h-Pseudo-Signature

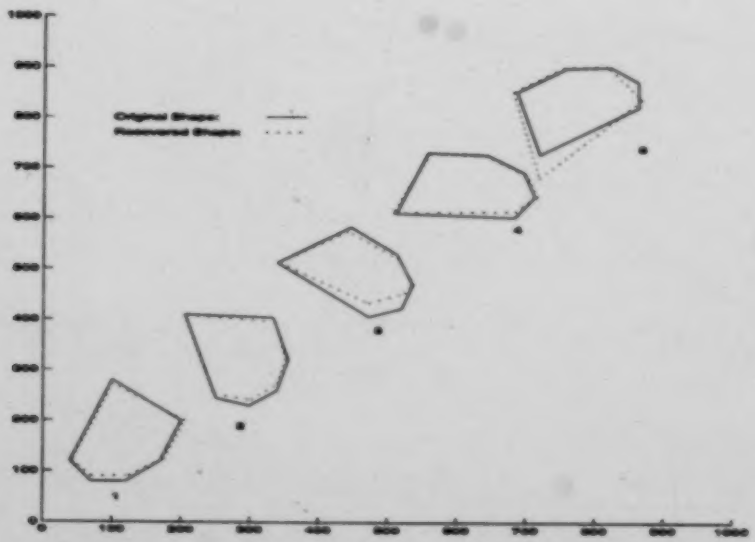


Figure 5.16: Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_h-Pseudo-Signature

Figure 5.17: Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_h-Pseudo-Signature

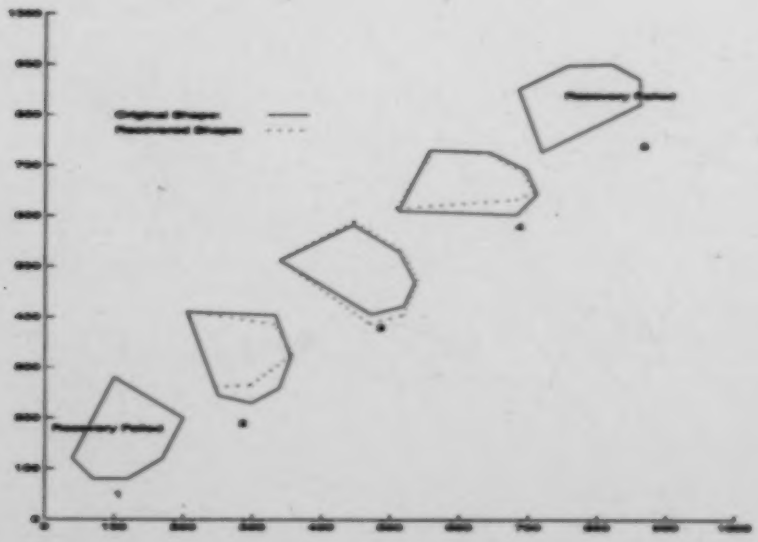


Figure 5.18: Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_h-Pseudo-Signature

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Vertex Reconstruction Signature Type: k_h Simulated Shape: Hexagon						
Reference		**Pos. 1**	**Pos. 2**	**Pos. 3**	**Pos. 4**	**Pos. 5**
Figure 5.15	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.16	M.C.	0.119	0.175	0.185	0.285	0.436
	A.O.E.	0.022	0.029	0.048	0.018	0.091
	A.P.E.	4.212	4.486	7.071	1.644	9.206
Figure 5.17	M.C.	0.217	0.403	0.477	0.489	0.651
	A.O.E.	0.075	0.050	0.058	0.035	0.125
	A.P.E.	3.872	7.513	6.874	6.115	14.638
Figure 5.18	M.C.	-	0.401	0.474	0.492	-
	A.O.E.	-	0.062	0.026	0.038	-
	A.P.E.	-	8.766	7.886	6.872	-

Table 5.4: Quantitative Mismatch between the Original and the Recovered Hexagon in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_h-Pseudo-Signature).

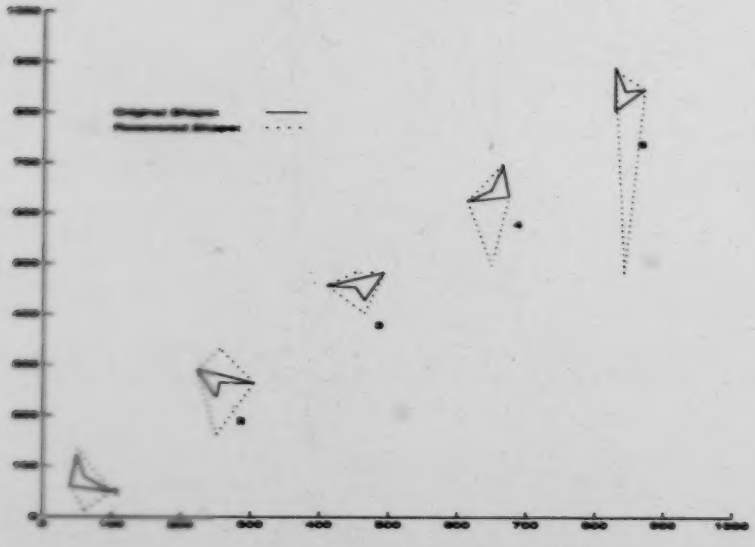


Figure 5.19: Recovering a Concave Polygon on an ideal CRS-CRS Sensor Using V.R. with k_h-Pseudo-Signature

5.4.2 Vertex Reconstruction with a CRS-CRS Hybrid Sensor using k_l-Pseudo-Signature

The fourth set of simulations was similar to the first, but based on the k_l-signature instead. In the first scenario, ideal condition was assumed. With perfect sensor output and transition detection in the pseudo-signatures, the values of k_l were used to recover the original shape perfectly (see Figure 5.20). In the second scenario, only C.Q.E. and T.E. was introduced. The maximum quantization error was $\pm 1/1000$ (the geometric resolution in the analog dimension) in each of the concerned columns. These quantization noises were directly reflected in the sensor output. In addition, T.E. was assumed to be ranged from -5 to $+5$ units with a uniform distribution. Figure 5.21 shows the corresponding shape recovery. Graphically, the recovered shapes were only slightly off from the original. In the next two scenarios, P.U. profile I and II were added on top of the C.Q.E. and T.E. to the sensor output. Their simulation results are depicted in Figure 5.22 and Figure 5.23 respectively. Table 5.5 shows the corresponding quantitative errors in each scenario. In general, the performance was still satisfactory with P.U. profiles I and II.

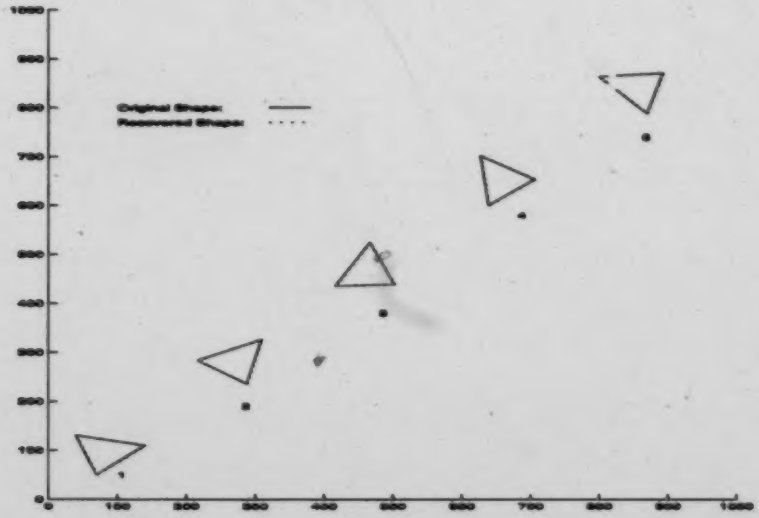


Figure 5.20: Recovering a Triangle on an ideal CRS-CRS Sensor by V.R. with k_l-Pseudo-Signature

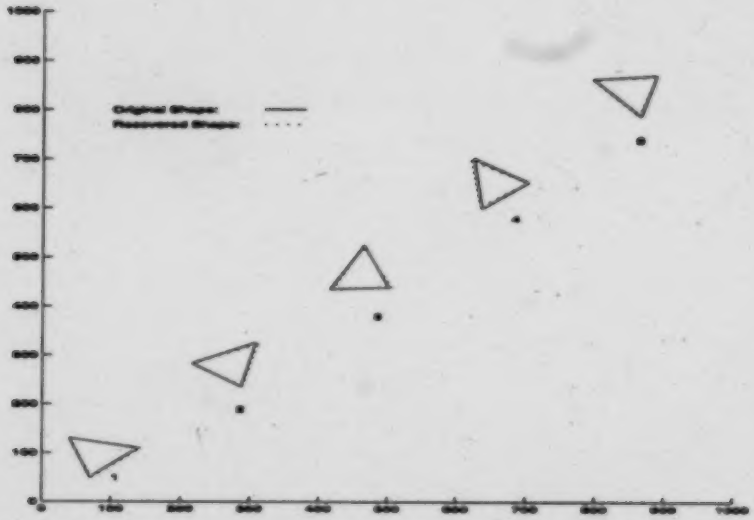


Figure 5.21: Recovering a Triangle on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_l-Pseudo-Signature

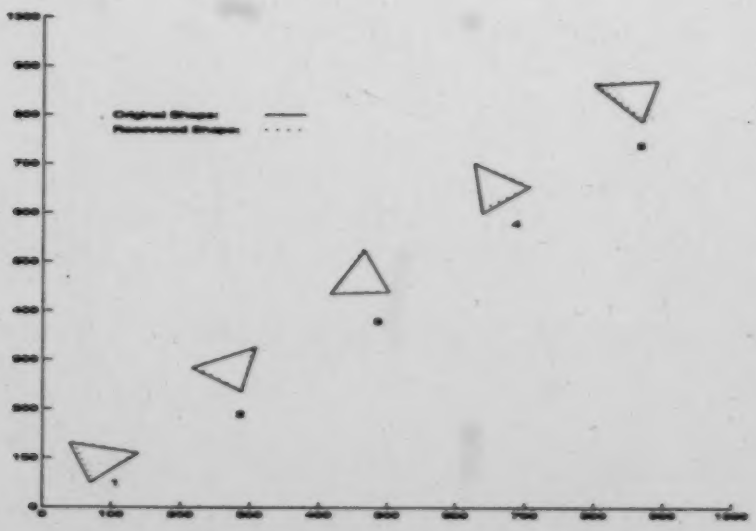


Figure 5.22: Recovering a Triangle on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_l-Pseudo-Signature

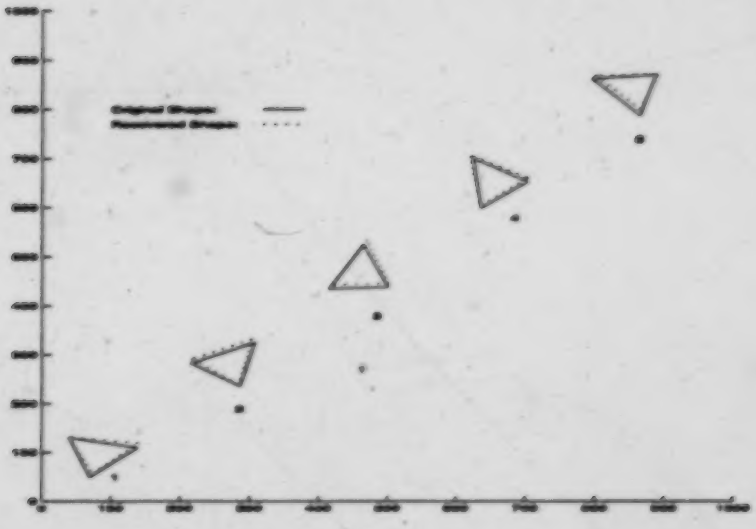


Figure 5.23: Recovering a Triangle on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_l-Pseudo-Signature

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Vertex Reconstruction Signature Type: k_l Simulated Shape: Triangle						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.20	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.21	M.C.	0.024	0.062	0.093	0.089	0.021
	A.O.E.	0.020	0.010	0.008	0.064	0.006
	A.P.E.	2.342	2.119	2.345	2.130	1.688
Figure 5.22	M.C.	0.052	0.059	0.080	0.086	0.055
	A.O.E.	0.007	0.008	0.011	0.014	0.009
	A.P.E.	2.167	1.032	1.240	2.024	1.935
Figure 5.23	M.C.	0.098	0.083	0.105	0.090	0.125
	A.O.E.	0.026	0.029	0.015	0.024	0.035
	A.P.E.	2.431	2.645	3.234	2.115	4.254

Table 5.5: Quantitative Mismatch between the Original and the Recovered Triangle in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_l-Pseudo-Signature).

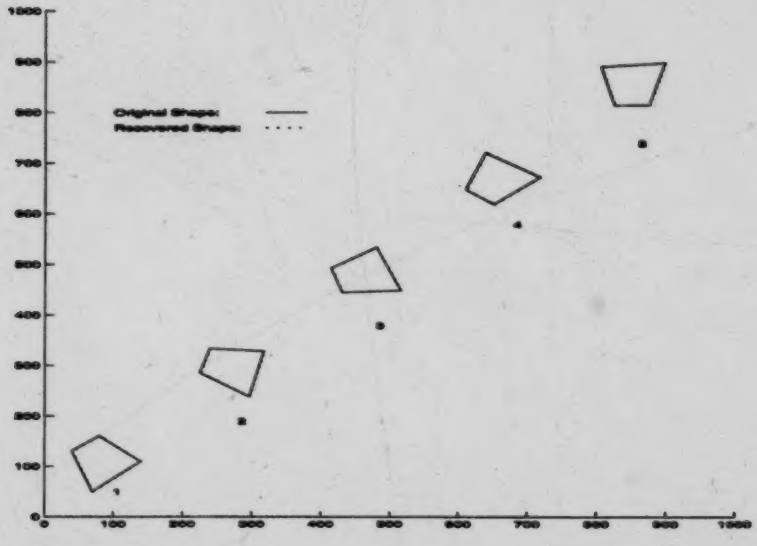


Figure 5.24: Recovering a Tetragon on an ideal CRS-CRS Sensor by V.R. with k_l-Pseudo-Signature

The fifth set of simulations were done with a tetragon. Except using k_l-signature, all other conditions are the same as the second set. Figure 5.24 to Figure 5.27 show the corresponding simulation results and Table 5.6 gives their quantitative errors. With ideal sensor output, Vertex Reconstruction recovered the original tetragon perfectly. When only C.Q.E. and T.E. were added, only insignificant errors were found in shape recovery. Even with P.U. profiles I and II, the tetragons could still be reconstructed with minor discrepancies.

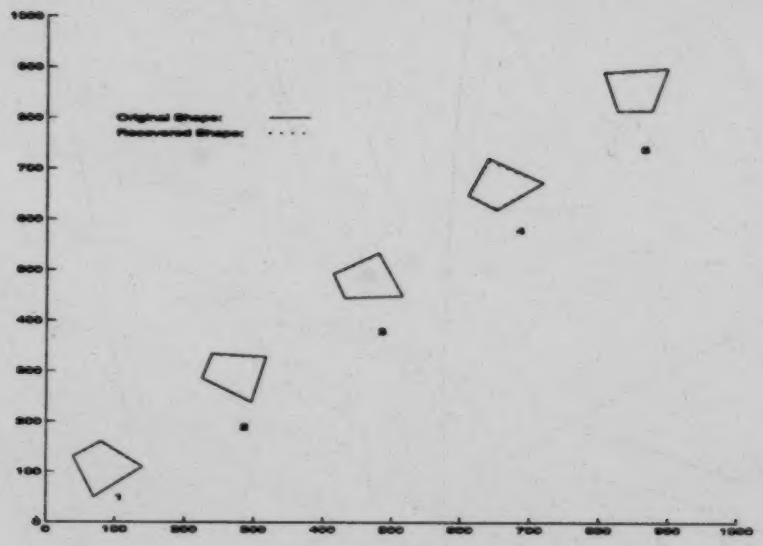


Figure 5.25: Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_l-Pseudo-Signature

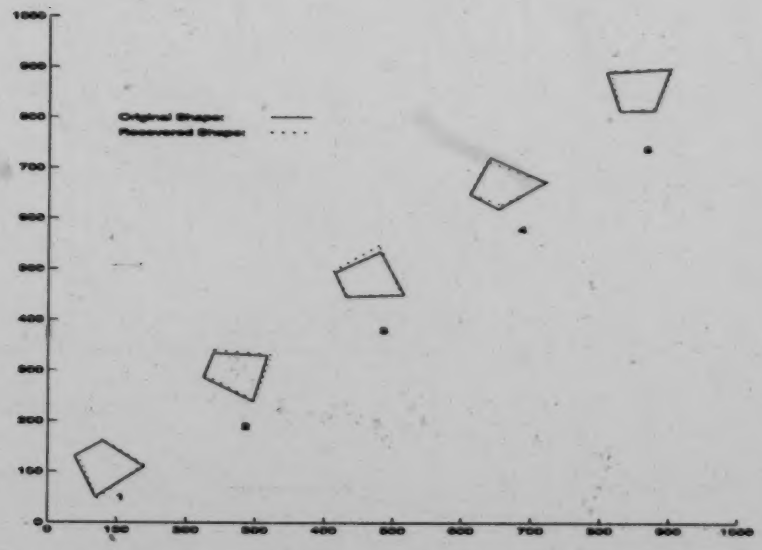


Figure 5.26: Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_l-Pseudo-Signature

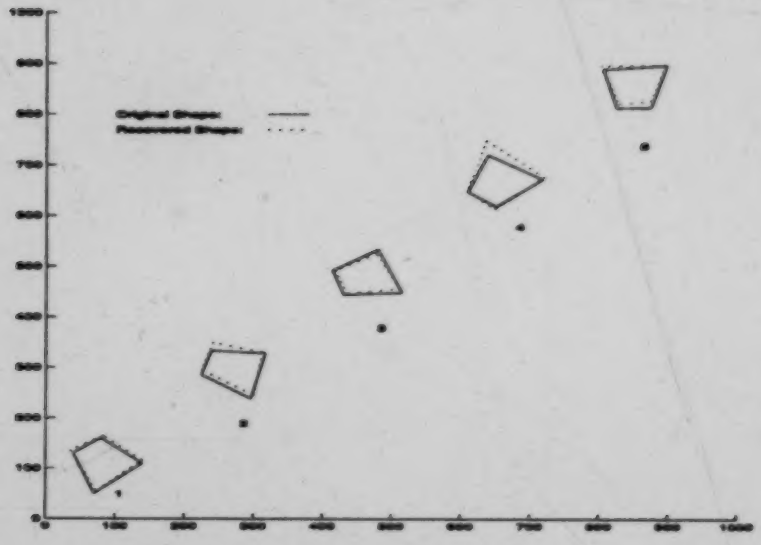


Figure 5.27: Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_l-Pseudo-Signature

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Vertex Reconstruction Signature Type: k_l Simulated Shape: Tetragon						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.24	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.25	M.C.	0.013	0.015	0.020	0.057	0.025
	A.O.E.	0.001	0.015	0.002	0.020	0.008
	A.P.E.	0.550	0.525	0.800	0.525	0.325
Figure 5.26	M.C.	0.080	0.109	0.137	0.125	0.117
	A.O.E.	0.067	0.060	0.035	0.032	0.010
	A.P.E.	1.578	2.091	3.687	1.897	1.923
Figure 5.27	M.C.	0.085	0.129	0.067	0.231	0.195
	A.O.E.	0.078	0.177	0.036	0.087	0.044
	A.P.E.	1.705	3.016	1.467	5.408	3.633

Table 5.6: Quantitative Mismatch between the Original and the Recovered Tetragon in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_l-Pseudo-Signature).

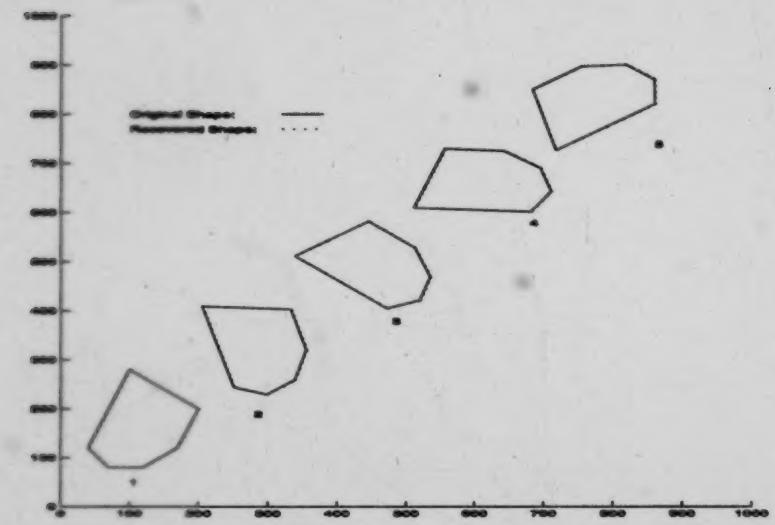


Figure 5.28: Recovering a Hexagon on an ideal CRS-CRS Sensor by V.R. with k_l-Pseudo-Signature

The sixth set of simulations was done with a hexagon. It was similar to the third set, except k_l-pseudo-signature was used instead. Figure 5.28 to Figure 5.31 show their corresponding simulation results under various error conditions. Table 5.7 summarizes their quantitative errors in each case. Similar to the results using a triangle and a tetragon, the algorithm reconstructed the original hexagon perfectly when an ideal sensor was simulated. Errors in shape recovery were still acceptable when C.Q.E. and T.E., and P.U. profile I were assumed. However, larger discrepancies in shape recovery were observed with P.U. profile II was used (see Figure 5.31). Since the dynamic range of k_h for a particular shape is usually small (from 0.1 to 0.2), it is difficult to extract transitional point from a seriously distorted signature. As the number of vertex increases, the chance of missing transitional points from the signature increases.

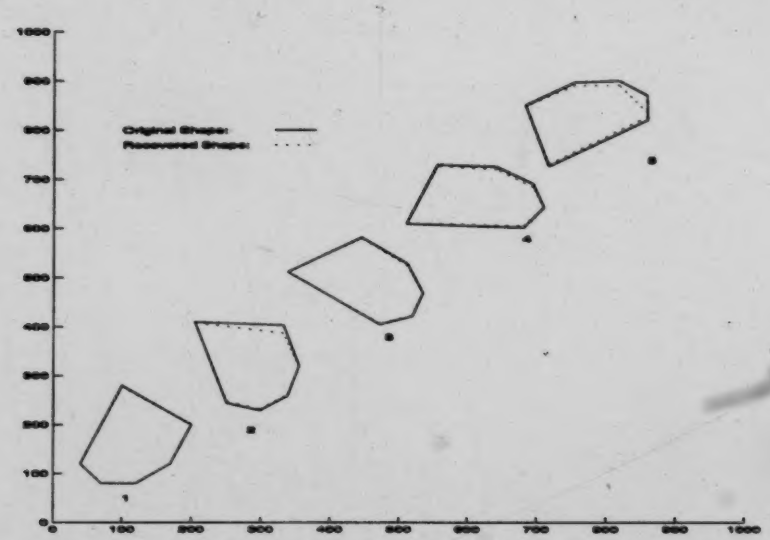


Figure 5.29: Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E. & T.E.) by V.R. with k_l-Pseudo-Signature

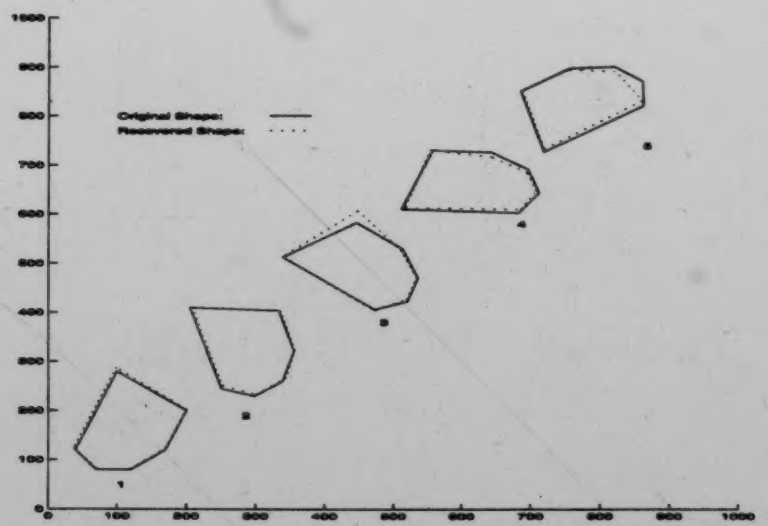


Figure 5.30: Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R. with k_l-Pseudo-Signature

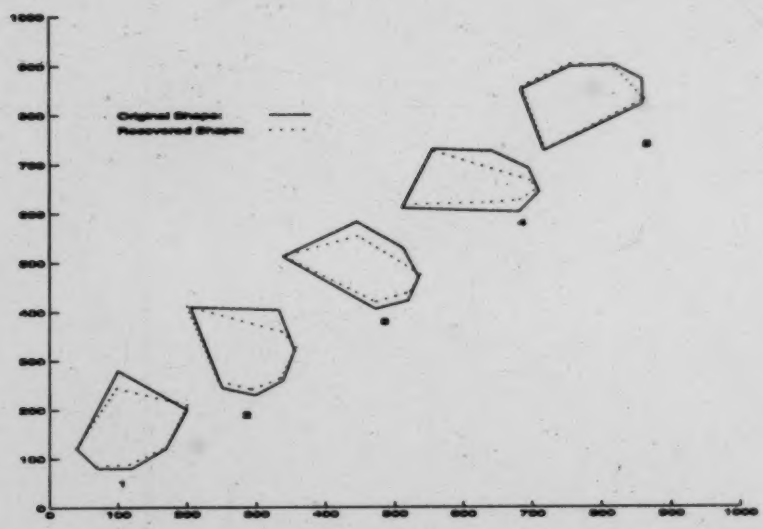


Figure 5.31: Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R. with k_l-Pseudo-Signature

As mentioned in Section 4.2.1, Vertex Reconstruction algorithm is only applicable on convex polygon. Figure 5.32 show the possible ambiguity it caused when the algorithm and k_l-signature is used on a concave shape.

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Vertex Reconstruction Signature Type: k_l Simulated Shape: Hexagon						
Reference		**Pos. 1**	**Pos. 2**	**Pos. 3**	**Pos. 4**	**Pos. 5**
Figure 5.28	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.29	M.C.	0.010	0.093	0.034	0.060	0.169
	A.O.E.	0.002	0.007	0.003	0.014	0.059
	A.P.E.	0.333	1.950	0.483	1.567	6.567
Figure 5.30	M.C.	0.030	0.028	0.195	0.079	0.172
	A.O.E.	0.012	0.008	0.068	0.020	0.060
	A.P.E.	1.983	2.307	6.304	2.451	5.607
Figure 5.31	M.C.	0.312	0.397	0.487	0.502	0.188
	A.O.E.	0.079	0.105	0.096	0.065	0.064
	A.P.E.	6.231	7.544	8.354	8.905	5.233

Table 5.7: Quantitative Mismatch between the Original and the Recovered Hexagon in Each Simulated Position (CRS-CRS Sensor and V.R. Algorithm with k_l-Pseudo-Signature).

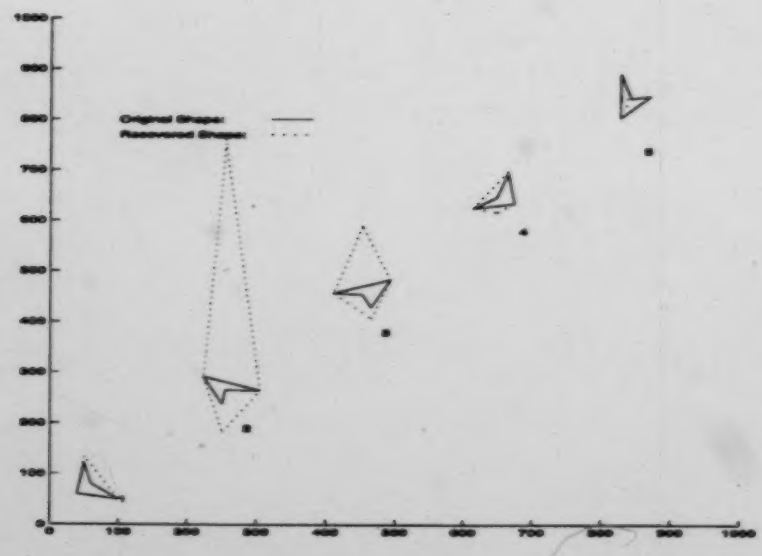


Figure 5.32: Recovering a Concave Polygon on an ideal CRS-CRS Sensor Using V.R. with k_l-Pseudo-Signature

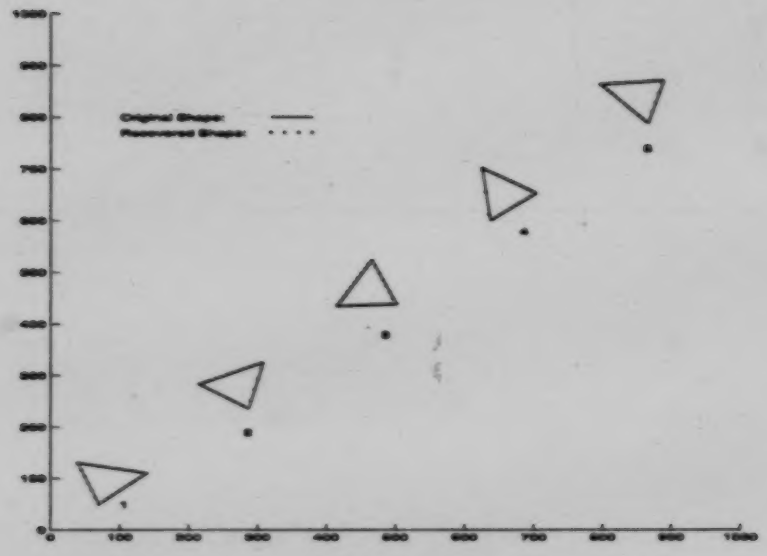


Figure 5.33: Recovering a Triangle on an ideal CRS-CRS Sensor by P.R.

5.4.3 Profile Reconstruction with a CRS-CRS Hybrid Sensor

With the choice of using both sets of k_h and k_l from a CRS-CRS hybrid sensor, Profile Reconstruction can be used for shape recovery. So the seventh set of simulations was done with a triangle on a CRS-CRS hybrid sensor using Profile Reconstruction. Figure 5.33 to Figure 5.36 show the simulation results under different error conditions. An ideal sensor was assumed in the first scenario which produced perfect result as expected. When only C.Q.E. was introduced in the second scenario, no significant error in shape recovery could be observed graphically (see Figure 5.34). Since Profile Reconstruction does not depend on the form of the pseudo-signature, T.E. was irrelevant in the simulations. As shown in Figure 5.35 and Figure 5.36, reasonably good shape recovery could be achieved. Table 5.8 summarizes the quantitative errors in shape recovery.

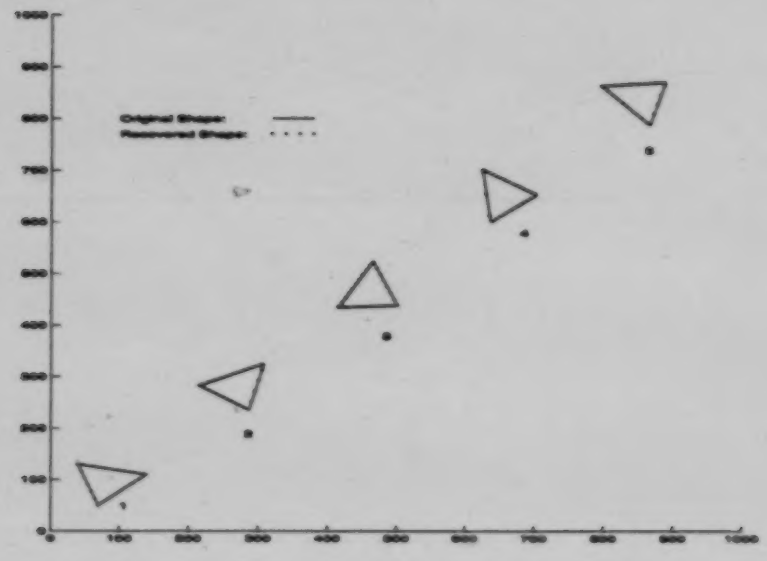


Figure 5.34: Recovering a Triangle on a CRS-CRS Sensor (C.Q.E.) by P.R.

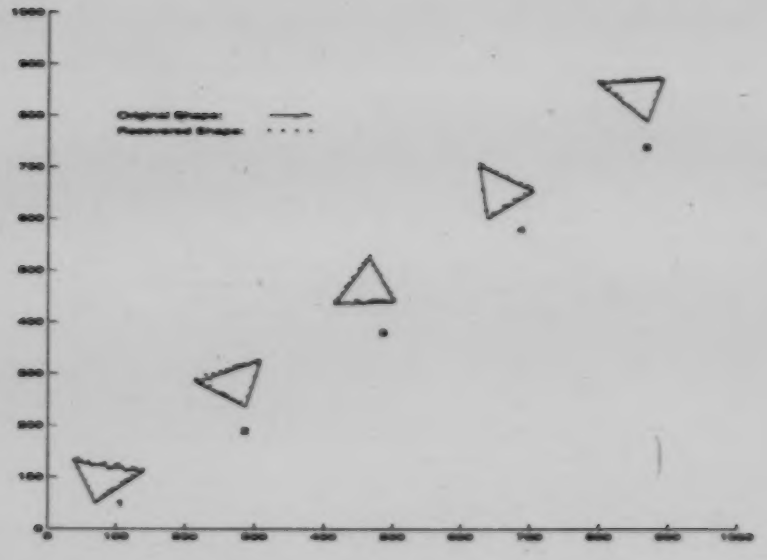


Figure 5.35: Recovering a Triangle on a CRS-CRS Sensor (C.Q.E. & P.U. Profile I) by P.R.

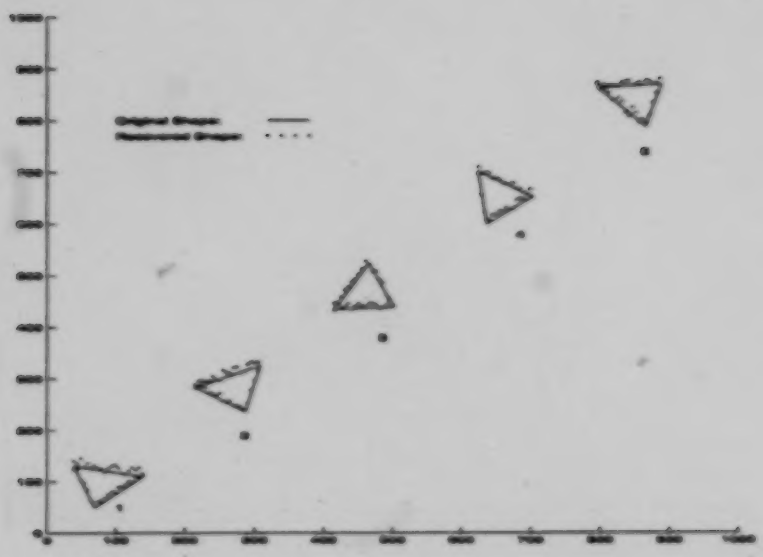


Figure 5.36: Recovering a Triangle on a CRS-CRS Sensor (C.Q.E. & P.U. Profile II) by P.R.

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Profile Reconstruction Simulated Shape: Triangle						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.33	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.34	M.C.	0.011	0.013	0.004	0.080	0.012
	A.O.E.	0.006	0.011	0.003	0.031	0.004
	A.P.E.	0.340	0.412	0.167	0.203	0.211
Figure 5.35	M.C.	0.024	0.035	0.064	0.051	0.029
	A.O.E.	0.014	0.021	0.020	0.015	0.022
	A.P.E.	4.318	2.517	3.084	2.341	2.305
Figure 5.36	M.C.	0.104	0.086	0.072	0.062	0.069
	A.O.E.	0.094	0.027	0.042	0.015	0.032
	A.P.E.	9.866	5.012	4.368	4.667	5.328

Table 5.8: Quantitative Mismatch between the Original and the Recovered Triangle in Each Simulated Position (CRS-CRS Sensor and P.R. Algorithm).

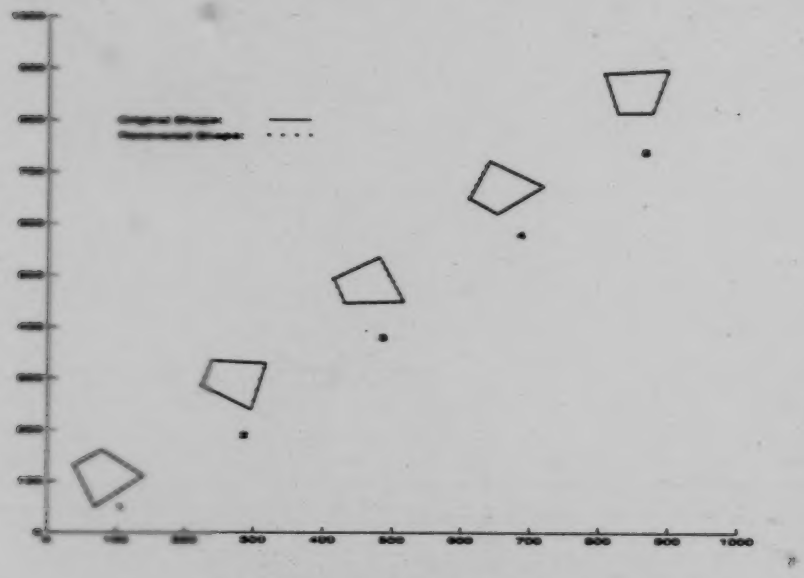


Figure 5.37: Recovering a Tetragon on an ideal CRS-CRS Sensor by P.R.

The eighth set of simulations was done with a tetragon on a CRS-CRS hybrid sensor using Profile Reconstruction. Figure 5.37 to Figure 5.40 show the simulation results under different error conditions. In the first scenario, the algorithm recovered the profile of the original tetragon perfectly when an ideal sensor was simulated. In the second scenario, there was no significant degradation in the accuracy of the algorithm in producing the original shape with the existence of C.Q.E. (see Figure 5.38). Shape recovery was acceptable with P.U. profiles I and II. Table 5.9 gives the quantitative errors in recovering a tetragon.

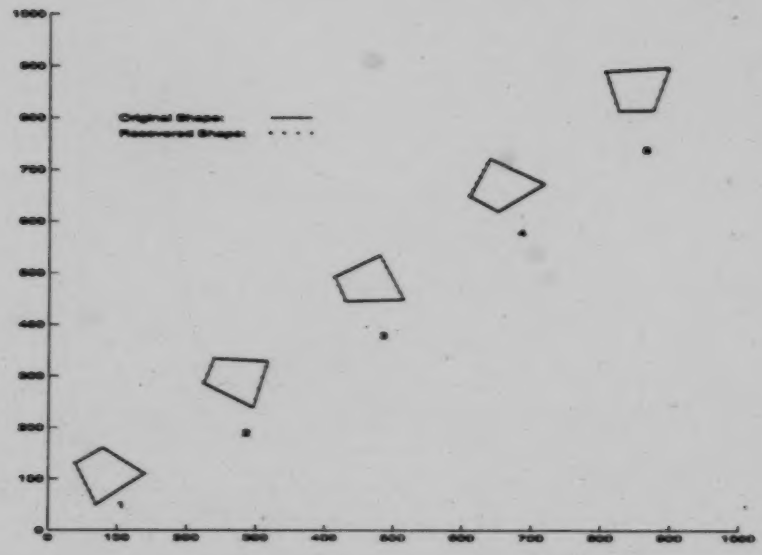


Figure 5.38: Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E.) by P.R.

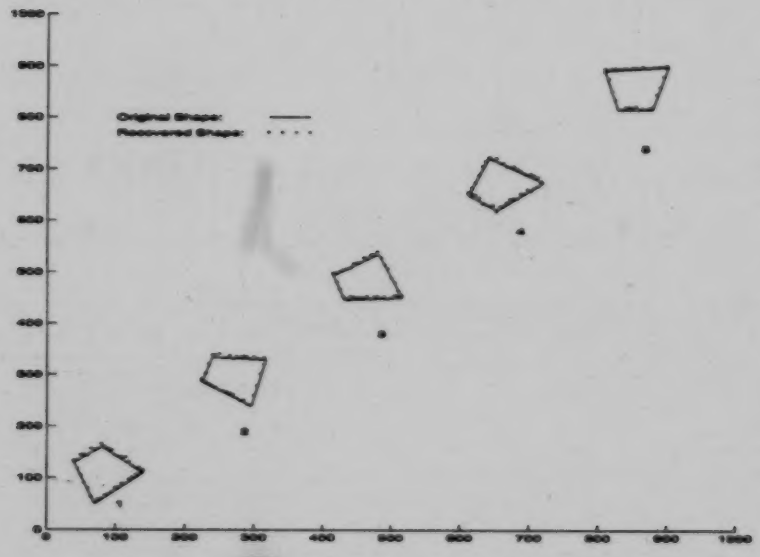


Figure 5.39: Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E. & P.U. Profile I) by P.R.

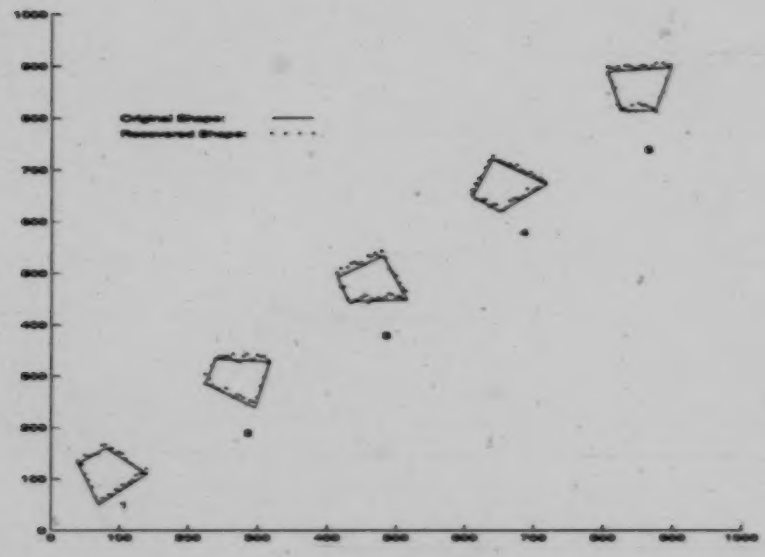


Figure 5.40: Recovering a Tetragon on a CRS-CRS Sensor (C.Q.E. & P.U. Profile II) by P.R.

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Profile Reconstruction Simulated Shape: Tetragon						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.37	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.38	M.C.	0.005	0.019	0.031	0.016	0.015
	A.O.E.	0.001	0.015	0.008	0.016	0.003
	A.P.E.	0.175	0.641	0.776	0.056	0.340
Figure 5.39	M.C.	0.064	0.045	0.048	0.052	0.040
	A.O.E.	0.009	0.022	0.010	0.019	0.008
	A.P.E.	4.238	2.515	2.614	2.331	2.508
Figure 5.40	M.C.	0.078	0.072	0.085	0.080	0.078
	A.O.E.	0.045	0.032	0.041	0.035	0.022
	A.P.E.	4.601	5.381	6.221	5.617	5.518

Table 5.9: Quantitative Mismatch between the Original and the Recovered Tetragon in Each Simulated Position (CRS-CRS Sensor and P.R. Algorithm).

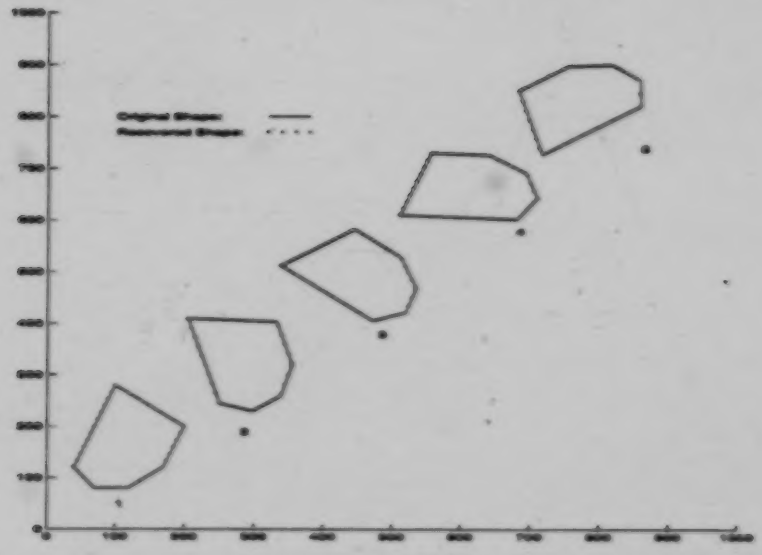


Figure 5.41: Recovering a Hexagon on an ideal CRS-CRS Sensor by P.R.

The ninth set of simulations was done with a hexagon on a CRS-CRS hybrid sensor using Profile Reconstruction. Figure 5.41 to Figure 5.44 show the simulation results with various noise levels. With an ideal sensor in the first scenario, the algorithm reproduced the profile of the original hexagon perfectly. In the second scenario, the algorithm still gave satisfactory results when only C.Q.E. was introduced (see Figure 5.42). The recovered profiles were acceptable with P.U. profiles I and II. Table 5.10 shows the quantitative errors in recovering a hexagon.

Figure 5.42: Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E.) by P.R.

Figure 5.43: Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E. & P.U. Profile I) by P.R.

Figure 5.44: Recovering a Hexagon on a CRS-CRS Sensor (C.Q.E. & P.U. Profile II) by P.R.

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Profile Reconstruction Simulated Shape: Hexagon						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.41	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.42	M.C.	0.006	0.023	0.015	0.032	0.034
	A.O.E.	0.009	0.001	0.004	0.008	0.022
	A.P.E.	0.050	0.644	0.480	0.264	2.750
Figure 5.43	M.C.	0.034	0.035	0.025	0.036	0.031
	A.O.E.	0.019	0.017	0.008	0.014	0.020
	A.P.E.	2.512	1.975	2.031	2.612	1.877
Figure 5.44	M.C.	0.062	0.073	0.075	0.070	0.085
	A.O.E.	0.025	0.030	0.021	0.022	0.018
	A.P.E.	3.624	2.931	3.678	4.213	3.031

Table 5.10: Quantitative Mismatch between the Original and the Recovered Hexagon in Each Simulated Position (CRS-CRS Sensor and P.R. Algorithm).

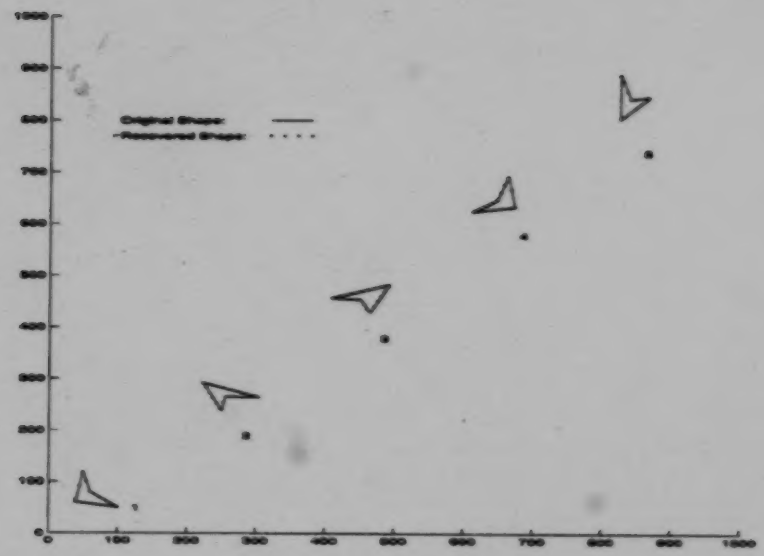


Figure 5.45: Recovering a Concave Tetragon on an ideal CRS-CRS Sensor by P.R.

As discussed in Chapter 4, the assumption of a straightly convex original shape is not necessary when the Profile Reconstruction Algorithm is used. As far as the convexity of the original shape is maintained in the digital dimension, the sets of k_h and k_l can still be used to estimate the original l and w in each sensor column. To verify this advantage of the Profile Reconstruction Algorithm, the tenth set of simulations used a concave tetragon. Figure 5.45 to Figure 5.48 show their graphical results. In the first scenario, an ideal sensor was assumed and the resulted profile matched the original perfectly (see Figure 5.45). Even when C.Q.E. was added in the second scenario, no significant mismatch between the recovered profile and the original could be observed in Figure 5.46. Satisfactory shape recovery could be achieved even under P.U. profiles I and II. Table 5.11 summarizes the quantitative error in shape recovery for the first three scenarios.

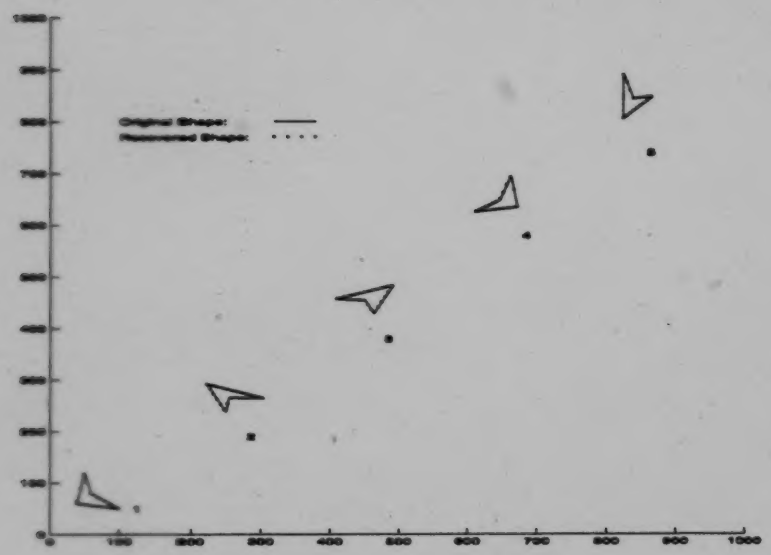


Figure 5.46: Recovering a Concave on a CRS-CRS Sensor (C.Q.E.) by P.R.

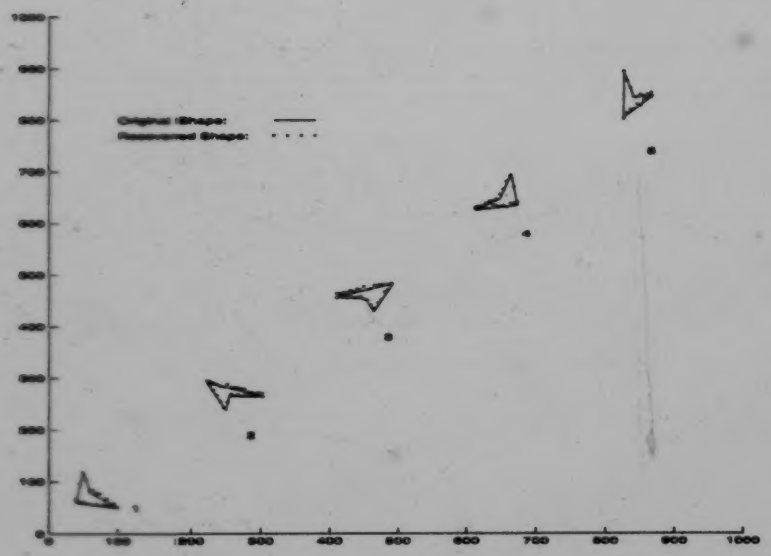


Figure 5.47: Recovering a Concave Tetragon on a CRS-CRS Sensor (C.Q.E & P.U. Profile I) by P.R.

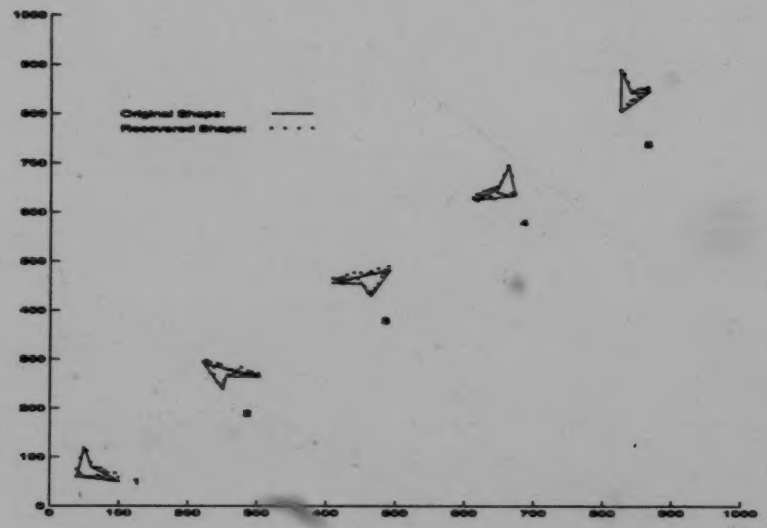


Figure 5.48: Recovering a Concave Tetragon on a CRS-CRS Sensor (C.Q.E & P.U. Profile II) by P.R.

Hybrid Sensor Type: CRS-CRS Recovery Algorithm: Profile Reconstruction Simulated Shape: Concave Tetragon						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.45	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.46	M.C.	0.037	0.035	0.021	0.042	0.036
	A.O.E.	0.033	0.014	0.005	0.010	0.014
	A.P.E.	1.575	0.378	0.437	0.382	0.395
Figure 5.47	M.C.	0.053	0.039	0.058	0.055	0.062
	A.O.E.	0.072	0.045	0.036	0.040	0.047
	A.P.E.	3.680	3.471	3.372	2.982	3.114
Figure 5.48	M.C.	0.072	0.080	0.077	0.098	0.095
	A.O.E.	0.080	0.067	0.058	0.069	0.072
	A.P.E.	4.831	3.902	5.011	4.367	3.584

Table 5.11: Quantitative Mismatch between the Original and the Recovered Concave Tetragon in Each Simulated Position (CRS-CRS Sensor and P.R. Algorithm).

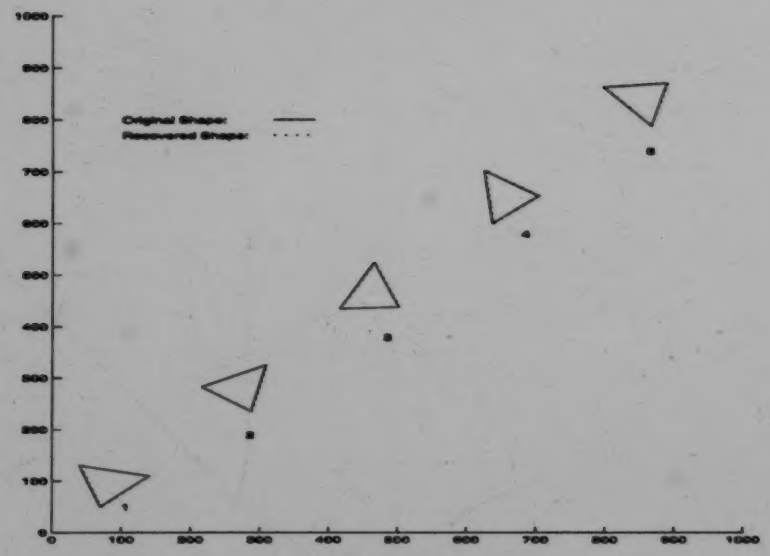


Figure 5.49: Recovering a Triangle on a ideal CRS-M Sensor by V.R.

5.5 Simulation Results with a CRS-M Hybrid Sensor

As only a set of k is provided by this type of sensor, we can only apply the Vertex Reconstruction algorithm for shape recovery. The eleventh set of simulations was done with a triangle on a CRS-M sensor using Vertex Reconstruction. Figure 5.49 to Figure 5.52 show the results of shape recovery under different error conditions and Table 5.12 summarizes the quantitative errors. In the first scenario, ideal condition was simulated and the resulting shape matched the original perfectly (see Figure 5.49). When only C.Q.E. and T.E. were included in the second scenario, the algorithm produced accurate estimates on the vertices as illustrated in Figure 5.50. In the third scenario, P.U. profiles I was added, yet only minor discrepancies were noted in the recovered shapes (see Figure 5.51). However, the performance of the algorithm got worse with increasing noise level. As shown in Figure 5.52, the recovered shapes were noisier with P.U. profile II.

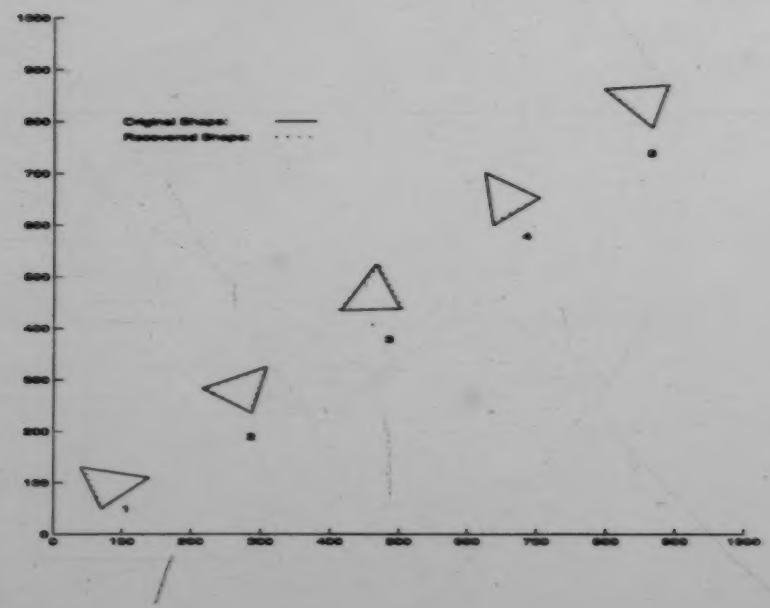


Figure 5.50: Recovering a Triangle on a CRS-M Sensor (C.Q.E. & T.E.) by V.R.

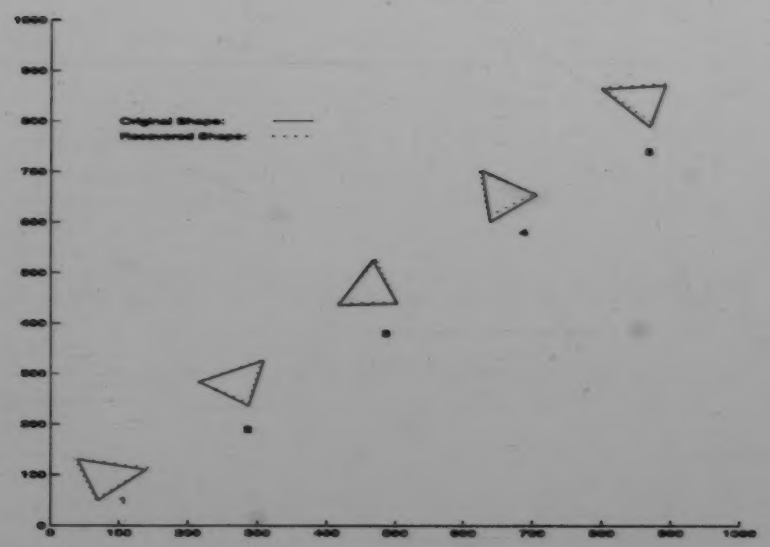


Figure 5.51: Recovering a Triangle on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R.

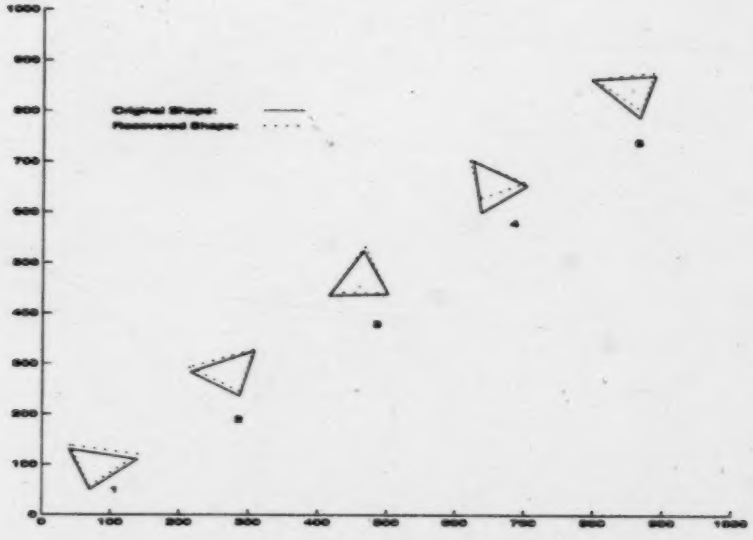


Figure 5.52: Recovering a Triangle on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R.

Hybrid Sensor Type: CRS-M Recovery Algorithm: Vertex Reconstruction Simulated Shape: Triangle						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.49	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.50	M.C.	0.046	0.060	0.074	0.051	0.030
	A.O.E.	0.029	0.009	0.064	0.018	0.003
	A.P.E.	1.020	0.448	1.044	1.407	1.609
Figure 5.51	M.C.	0.055	0.072	0.063	0.088	0.076
	A.O.E.	0.011	0.023	0.030	0.056	0.043
	A.P.E.	2.315	2.612	1.981	3.207	3.650
Figure 5.52	M.C.	0.082	0.098	0.075	0.198	0.093
	A.O.E.	0.045	0.036	0.035	0.117	0.053
	A.P.E.	5.373	4.602	3.681	10.237	5.314

Table 5.12: Quantitative Mismatch between the Original and the Recovered Triangle in Each Simulated Position (CRS-M Sensor and V.R. Algorithm).

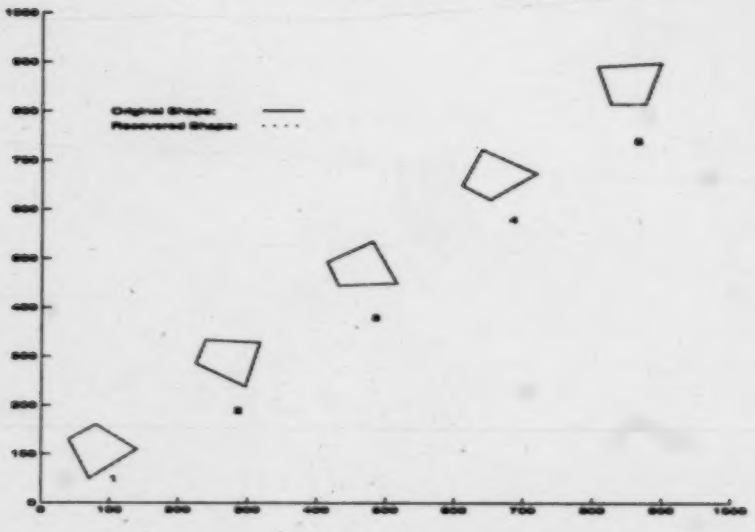


Figure 5.53: Recovering a Tetragon on a ideal CRS-M Sensor by V.R.

The twelfth set of simulations was done on a tetragon. Figure 5.53 to Figure 5.56 show graphically the results with various error conditions and Table 5.13 summarizes their quantitative errors. With an ideal CRS-M sensor, Vertex Reconstruction recovered the original tetragon perfectly. When C.Q.E. and T.E. were introduced, only insignificant errors were found in shape recovery. With P.U. profile I, the recovered shapes were only slight off from the original. The error in shape recovery increased as P.U. profile II is used.

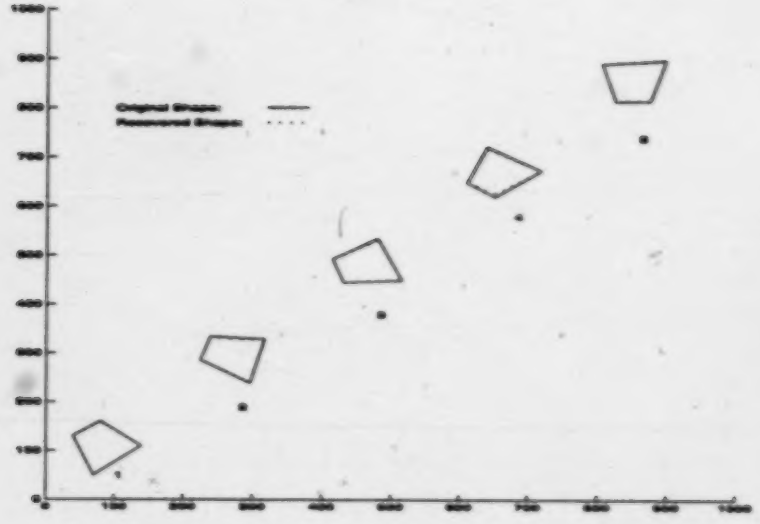


Figure 5.54: Recovering a Tetragon on a CRS-M Sensor (C.Q.E. & T.E.) by V.R.

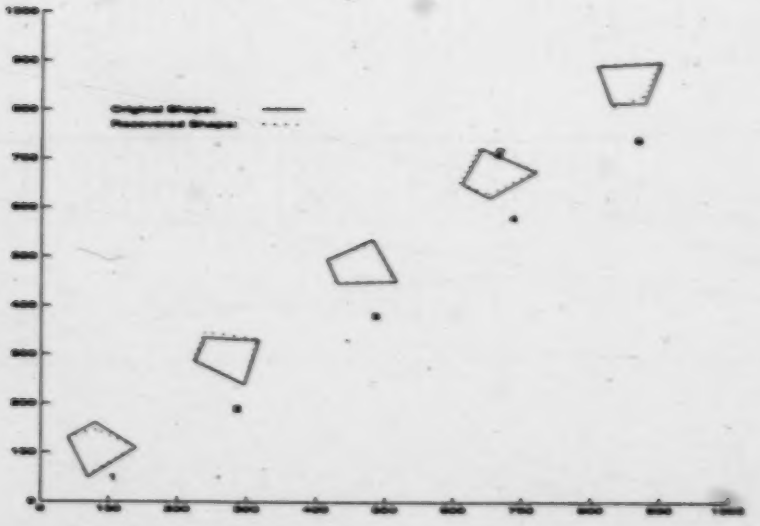


Figure 5.55: Recovering a Tetragon on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R.

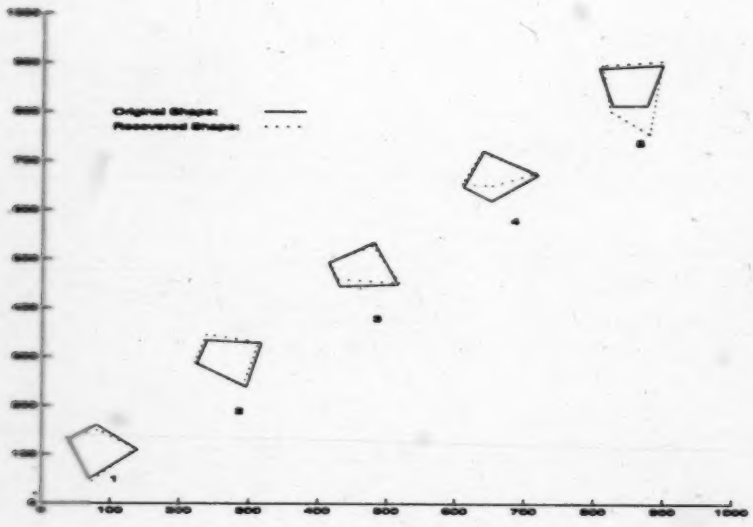


Figure 5.56: Recovering a Tetragon on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R.

Hybrid Sensor Type: CRS-M Recovery Algorithm: Vertex Reconstruction Simulated Shape: Tetragon						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.53	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.54	M.C.	0.014	0.022	0.028	0.082	0.041
	A.O.E.	0.001	0.026	0.004	0.000	0.006
	A.P.E.	0.525	0.625	0.600	1.300	0.350
Figure 5.55	M.C.	0.083	0.052	0.065	0.076	0.107
	A.O.E.	0.48	0.014	0.010	0.024	0.050
	A.P.E.	5.112	4.338	0.781	3.538	4.611
Figure 5.56	M.C.	0.078	0.060	0.081	0.128	0.312
	A.O.E.	0.045	0.020	0.063	0.084	0.103
	A.P.E.	4.981	4.211	6.225	15.367	20.328

Table 5.13: Quantitative Mismatch between the Original and the Recovered Tetragon in Each Simulated Position (CRS-M Sensor and V.R. Algorithm).

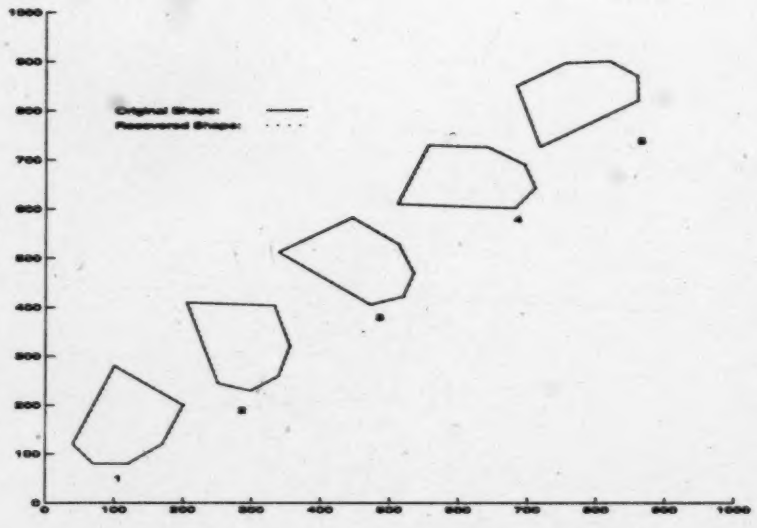


Figure 5.57: Recovering a Hexagon on a ideal CRS-M Sensor by V.R.

The thirteenth set of simulations was done with a hexagon on a CRS-M sensor using Vertex Reconstruction for shape recovery. Figure 5.57 to Figure 5.60 show the simulated results under various error conditions. An ideal sensor was simulated in the first scenario, the algorithm was able to recover the original hexagon perfectly (see Figure 5.57). C.Q.E. and T.E. were then added in the second scenario, only minor errors were found in shape recovery as shown in Figure 5.58. When P.U. profiles I and II were added, noisy signatures caused inaccurate determination of transitional points. Table 5.14 summarizes their quantitative errors.

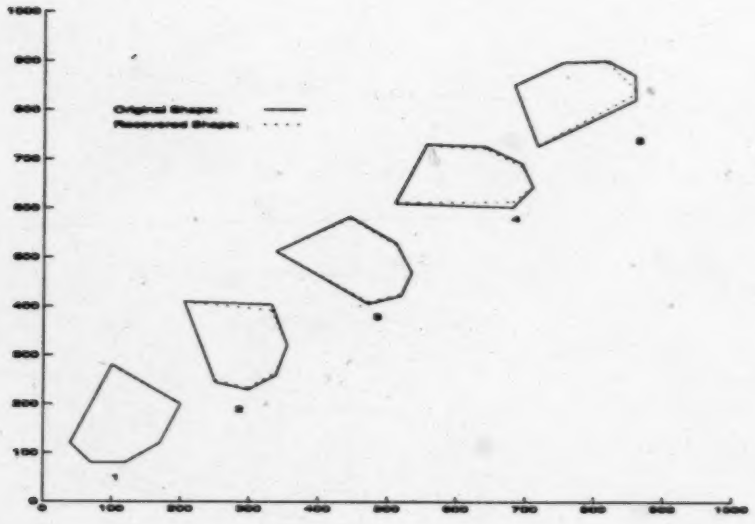


Figure 5.58: Recovering a Hexagon on a CRS-M Sensor (C.Q.E. & T.E.) by V.R.

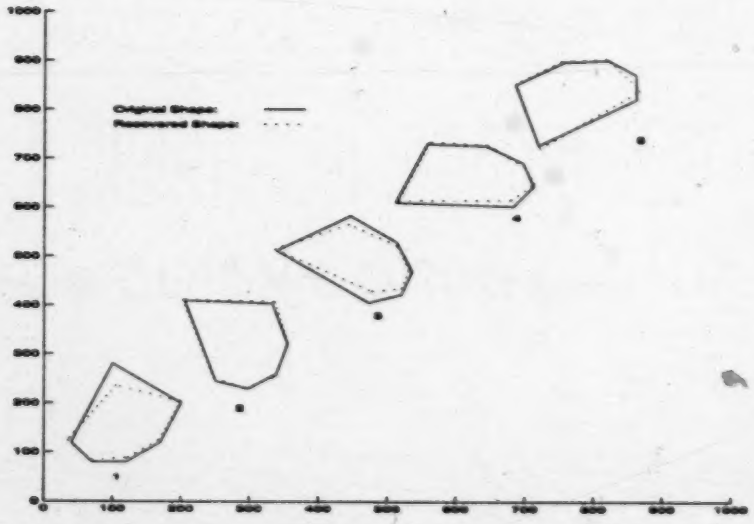


Figure 5.59: Recovering a Hexagon on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile I) by V.R.

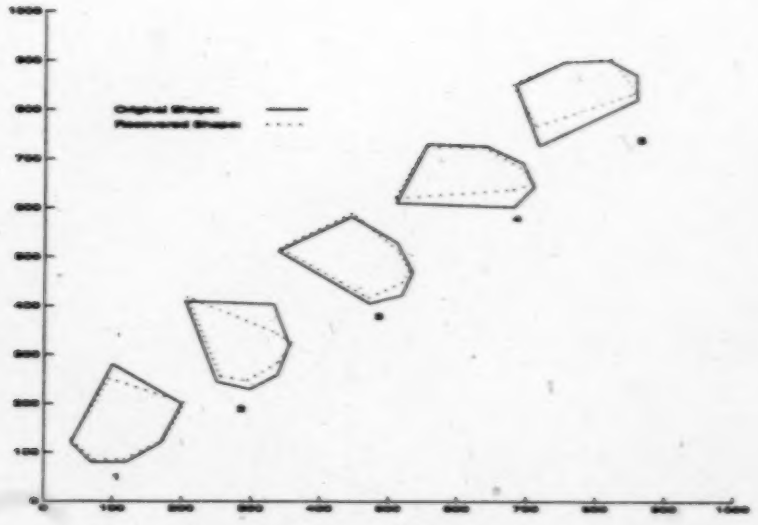


Figure 5.60: Recovering a Hexagon on a CRS-M Sensor (C.Q.E., T.E. & P.U. Profile II) by V.R.

Hybrid Sensor Type: CRS-M Recovery Algorithm: Vertex Reconstruction Simulated Shape: Hexagon						
Reference		Pos. 1	Pos. 2	Pos. 3	Pos. 4	Pos. 5
Figure 5.57	M.C.	0.000	0.000	0.000	0.000	0.000
	A.O.E.	0.000	0.000	0.000	0.000	0.000
	A.P.E.	0.000	0.000	0.000	0.000	0.000
Figure 5.58	M.C.	0.016	0.074	0.042	0.192	0.231
	A.O.E.	0.000	0.001	0.000	0.005	0.029
	A.P.E.	0.133	0.233	0.133	0.350	2.600
Figure 5.59	M.C.	0.320	0.042	0.361	0.187	0.205
	A.O.E.	0.107	0.015	0.048	0.036	0.045
	A.P.E.	13.751	1.687	5.371	2.652	3.065
Figure 5.60	M.C.	0.389	0.325	0.367	0.426	0.438
	A.O.E.	0.097	0.105	0.121	0.114	0.053
	A.P.E.	12.652	15.311	4.365	7.231	10.318

Table 5.14: Quantitative Mismatch between the Original and the Recovered Hexagon in Each Simulated Position (CRS-M Sensor and V.R. Algorithm).

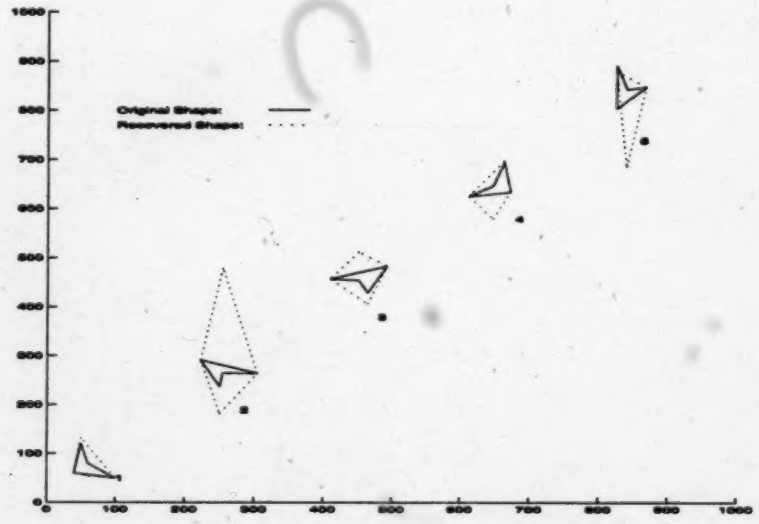


Figure 5.61: Recovering a Concave Polygon on an ideal CRS-M Sensor Using V.R. with k-Pseudo-Signature

As discussed in Chapter 4, the Vertex Reconstruction Algorithm failed to interpret concave shape. Figure 5.61 illustrates the typical ambiguity it caused even when an ideal sensor output was assumed.

5.6 Summary of Simulations

The simulations verified both the Vertex Reconstruction and Profile Reconstruction algorithms. Under ideal conditions, they worked perfectly in recovering the contact shapes. It is shown that Vertex Reconstruction must assume convex polygonal shape, otherwise erroneous shapes were recovered (see Figure 5.19, Figure 5.32 and Figure 5.61). On the other hand, Profile Reconstruction was able to recover those convex polygons simulated for Vertex Reconstruction as well as the concave shapes with the concavity occurred in the digital dimension of the sensor (see Figure 5.45 to Figure 5.48). Vertex Reconstruction depends on the transitional points of a pseudo-signature to recover the vertices of the polygonal shape. It was found that as noise was introduced, pseudo-signatures were distorted. When the noise level was high, P.U. profile II for instance, it was impossible to locate any reliable transitional points for certain configurations of a shape. On the other hand, Pro-

file Reconstruction estimates the lower and upper contact points of each sensing element, which is independent on the outputs from other sensing elements. It is only affected by the error from the concerned sensing element and no transitional point detection is required. Besides, the whole profile of the contact shape was recovered, which is more precise than just the vertices produced by Vertex Reconstruction. As a result, the performance of Profile Reconstruction was better than that of Vertex Reconstruction.

With Vertex Reconstruction, there are choices of k_h, k_l or k, depending on whether a CRS-CRS sensor or CRS-M sensor is used. It was found that under noisy conditions, shape recovery using k_l or k produced similar quantitative errors. Under similar noise conditions, the performance of Vertex Reconstruction using k_h was inferior to that of using either k_l or k. Though it is difficult to compare all the shape recovery case by case, the difference in the general form of coefficients in matrix A and B (see Section 4.2.4) provides some explanations on this phenomenon. In setting up the system of linear equations using k_h, all the coefficients contain the factor k_h. However, in the case of using k_l or k, some of the coefficients are constants such as $2L$ or L, which do not have the factors k_l or k. Hence, the effect of error in k_l or k on the solution of their corresponding system of equations is less serious than the same amount of error in k_h on the solution of its corresponding system of equations.

Chapter 6

Experimental Results

From the simulation results in Chapter 5, it is noted that the shape recovery algorithms are sensitive to noise in the sensor output. Shape recovery becomes unreliable when P.U. profile II is simulated. To see what level of P.U. can be obtained in practice, three prototypes of hybrid sensor were made and some experimental data were collected for analysis.

6.1 Experiments with a CRS-CRS Sensor

The prototype of a CRS-CRS hybrid sensor is shown in Figure 6.1. Conductive silicone rubber of 1 mm thickness was used. It is commercially available as RFI/EMI shielding materials [18] and comes in the form of a sheet. To use it in the prototype, 3 mm wide strips were sliced from the rubber sheet. As the conductivity of silicone rubber changed when it was stretched, the linear conductivity of each strip was adjusted before fixing the two end points. Ten pairs of strips were aligned in parallel and a hard board was used as backing. Non-conductive fabric mesh was laid between the top layer and the bottom layer of strips. These two layers of strips did not touch when no force was applied. When a force was applied on the surface of the sensing elements, their top strips touched the corresponding strips at the bottom through the openings of the fabric mesh. The setup measures 6.5 cm (digital dimension) by 32.5 cm (analog dimension). Since there were only ten columns of sensing elements in the setup, the resolution in the digital dimension was limited to 1.5 sensing element per cm. As visual measurement was done in the analog dimension, its resolution was 0.1 cm, which was limited by the tape measure used. Since the length of each sensing element was 32.5 cm, the equivalent maximum C.Q.E. was $\pm\frac{1}{325}$. For simplicity, only a rectangular object was used in the experiments.

Figure 6.1: Prototype of a CRS-CRS Sensor with Ten Sensing Elements

6.1.1 Calibrations of a CRS-CRS Sensor

As discussed in Chapter 3, it is ideal when conductivities of the two rubber strips in a pair are assumed to be equal and linearly uniform. In practice, ideal conditions are difficult to achieve and calibration is required to compensate major discrepancies. First, each sensor column was equally divided into 32.5 units along the length of the strips. Then, three runs of calibrations with a reference voltage of 6.05V were done as follows:

1. A point contact was made at the 1, 2, 3, ... unit mark in turn. Both V_h and V_l were recorded. They were compared to the expected values according to equations 3.1 and 3.2. Their means were then calculated as $V_{h(w=0)}$ and $V_{l(w=0)}$.
2. A 5-unit long contact was made starting at the 1, 2, 3, ... unit marks. Another pair of means, $V_{h(w=5)}$ and $V_{l(w=5)}$ were obtained.
3. Repeat step 2 above with a 10-unit length of contact and obtain the means, $V_{h(w=10)}$ and $V_{l(w=10)}$.

Figure 6.2 to Figure 6.4 show the calibration graphs for the first strip-pair of the CRS-CRS sensor prototype. It was found that the measured voltages (represented by crosses in the figures) were consistently higher than the expected voltages (represented by solid line in the figures). This discrepancy various slightly with w for a given l. It was impractical and impossible to tabulate the required voltage adjustment for each combination of l and w.

$V_{ref} = 6.05$V	Sensing Element Number									
	1	2	3	4	5	6	7	8	9	10
Calibration Factor for V_h (V)	0.345	0.016	0.353	0.317	0.080	0.161	0.070	0.261	0.168	0.291
Calibration Factor for V_l (V)	0.316	0.006	0.359	0.325	0.073	0.155	0.072	0.243	0.137	0.290

Table 6.1: Calibration Factors for Each Sensing Element of the CRS-CRS Hybrid Sensor Prototype.

Instead, the mean difference between the measured and expected voltages from the above calibration results was used as an calibration factor as follows:

$$V_{hOffset} = \frac{V_{h(w=0)} + V_{h(w=5)} + V_{h(w=10)}}{3} \tag{6.1}$$

$$V_{lOffset} = \frac{V_{l(w=0)} + V_{l(w=5)} + V_{l(w=10)}}{3} \tag{6.2}$$

Table 6.1 lists the calibration factors for all the ten sensing elements. In practice, the corresponding calibration factor was subtracted from the measured voltage of each sensing element before further processing. For example, if the measured outputs are V_h and V_l, the adjusted outputs are given by:

$$V_{hAdj} = V_h - V_{hOffset} \tag{6.3}$$

$$V_{lAdj} = V_l - V_{lOffset} \tag{6.4}$$

Though this calibration is simple, it is a compromise by approximating a non-linear function with a linear function. Over-adjustment exists near the two ends of a sensing element, but it worked reasonably well with the sensor prototype.

6.1.2 Results of Recovering a Rectangle Using the Vertex Reconstruction Algorithm with a CRS-CRS Sensor and k_h-Pseudo-Signature

Experiments were done with a rectangular planar object of dimension 5 cm by 3.5 cm. It was pressed against different positions on the sensor surface. Typical results are illustrated

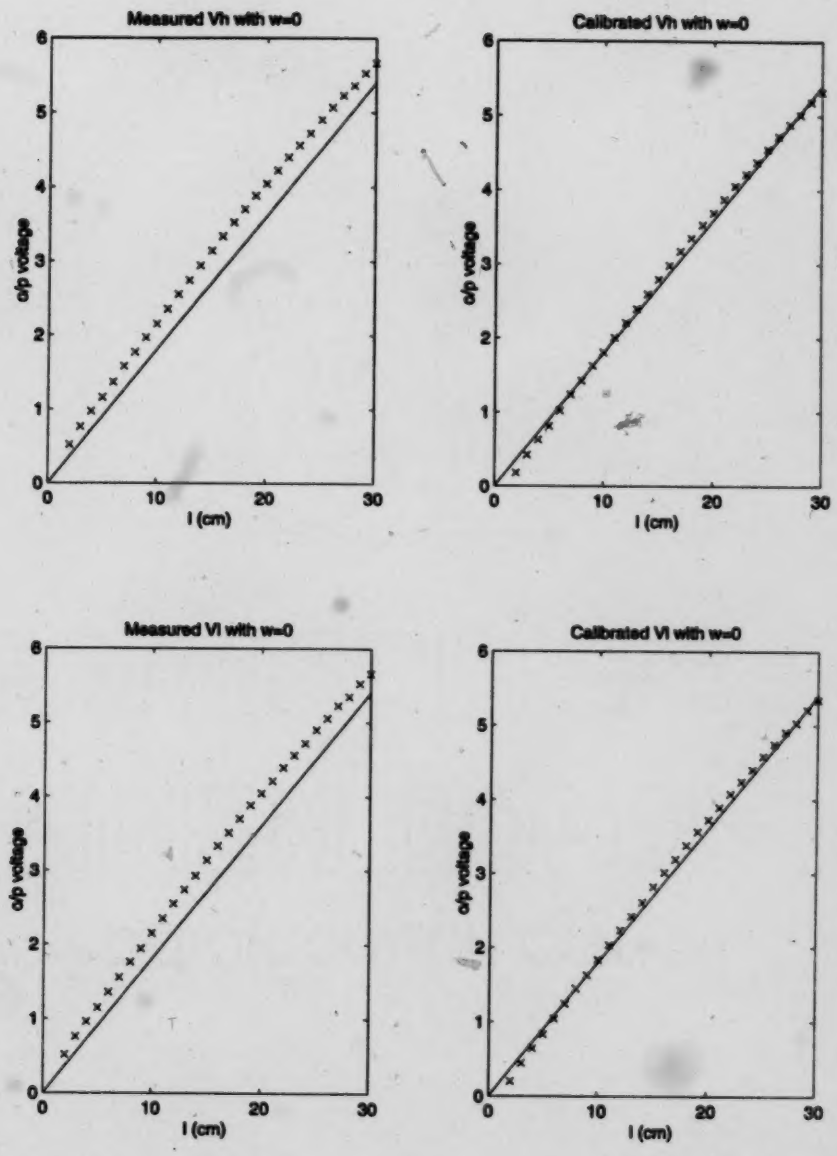


Figure 6.2: Calibration Graphs for the First Sensing Element in the CRS-CRS Hybrid Sensor Prototype ($w = 0$).

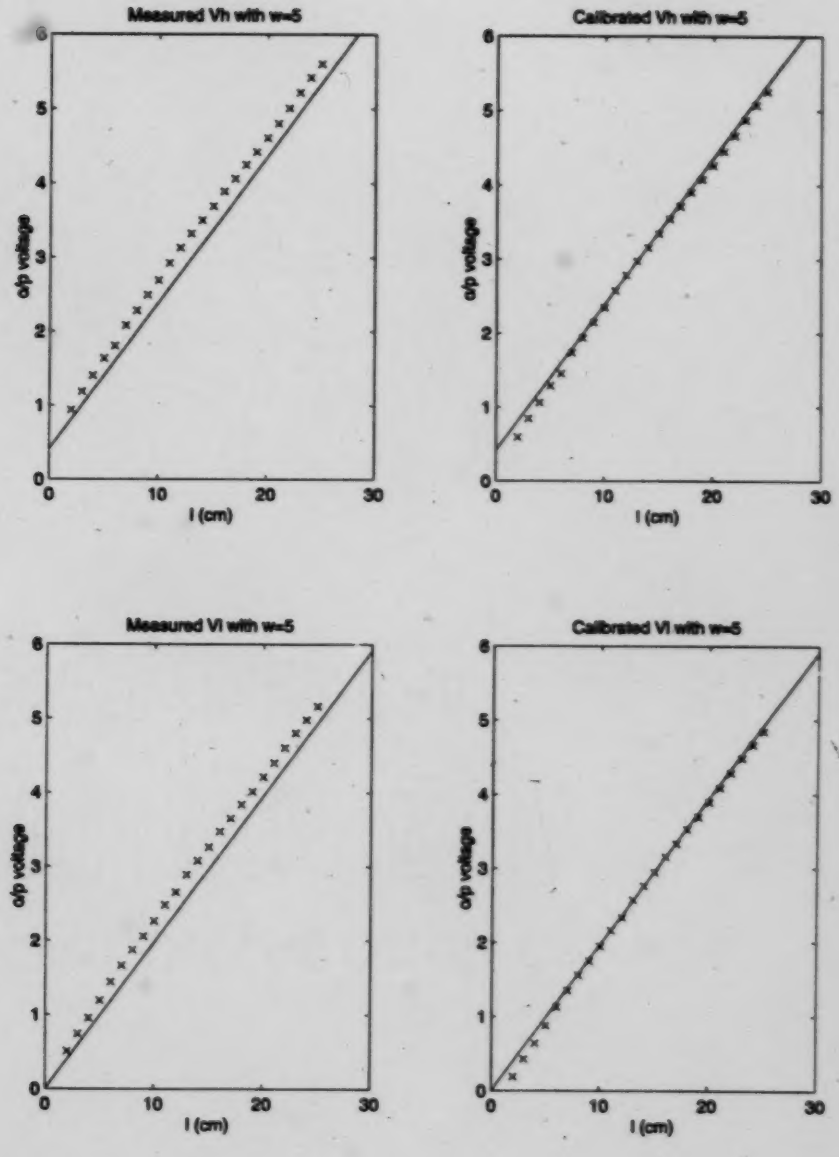


Figure 6.3: Calibration Graphs for the First Sensing Element in the CRS-CRS Hybrid Sensor Prototype ($w = 5$).

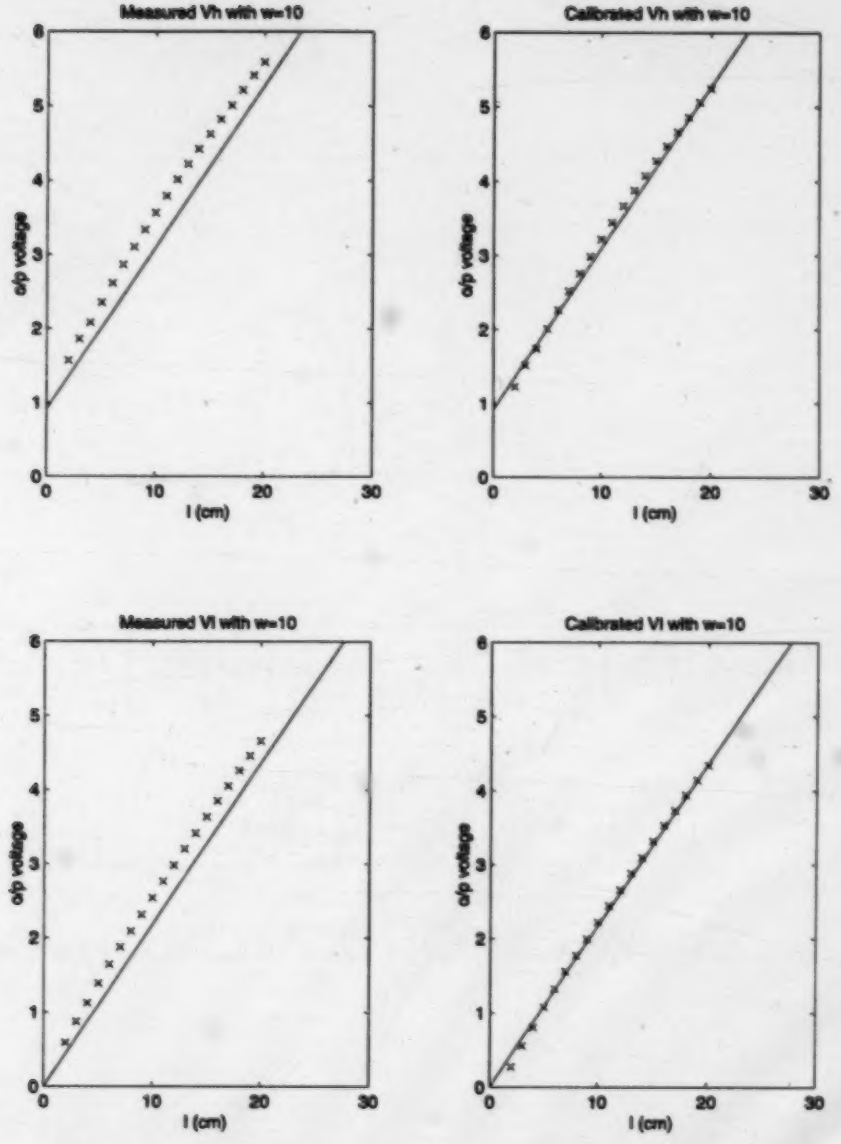


Figure 6.4: Calibration Graphs for the First Sensing Element in the CRS-CRS Hybrid Sensor Prototype ($w = 10$).

	Figure 6.5	Figure 6.6	Figure 6.7	Figure 6.9	Figure 6.10
M.C. (Radian-cm)	0.692	0.985	0.574	0.969	0.996
A.O.E. (Radian)	0.385	0.307	0.116	0.139	0.621
A.P.E. (cm)	1.545	2.925	0.525	0.510	1.617

Table 6.2: Quantitative Mismatch Between the Original and the Recovered Rectangle in Each Experiment (CRS-CRS Sensor Prototype and V.R. Algorithm with k_h-Signature).

in Figure 6.5 to Figure 6.10. In Figure 6.5, the object was put on the lower region of the CRS-CRS sensor. The transitions in the corresponding pseudo-signature was not prominent enough for accurate detection. Moreover, relatively large error in k_h resulted from the fourth sensing element. Hence, the recovered shape significantly deviated from the original. Figure 6.6 is the result of using raw data without calibration. Only slight improvement was noted with calibration. In Figure 6.7, the object was placed on the middle region of the sensor surface. The transition at the seventh sensing element was not distinctive either. However, the measured values of k_h were good enough to give a reasonably recovered shape. In this case, calibration was important to obtain a good signature. Figure 6.8 shows the signature without calibration. It is practically impossible to extract any transitional points. In Figure 6.9, the object was pressed against the upper region of the sensor. Similar to the previous experiments, the transition at the seventh sensing element was not prominent. Hence relatively large discrepancy between the corresponding recovered vertex and the original was produced. However, this result is still better that the one without calibration (see Figure 6.10). Table 6.2 lists the quantitative errors of the recovered shape.

From these experiments, it was noted that shape recovery was easily affected by output noise at the lower region of the sensor. The magnitude of output was relatively small when an object was located near the lower region. Since a mean calibration factor was adopted for output correction, over-adjustment might be done. On the contrary, more accurate output correction was done when the object was put on the middle and upper region of the sensor. Hence the reconstructed vertices were closer to the original. At these regions, the determining factor on the success of the Vertex Reconstruction algorithm was the accuracy in detecting the transitions in a pseudo-signature. Besides, calibration is important to give a properly aligned signature irrespective to different offsets on each sensing element.

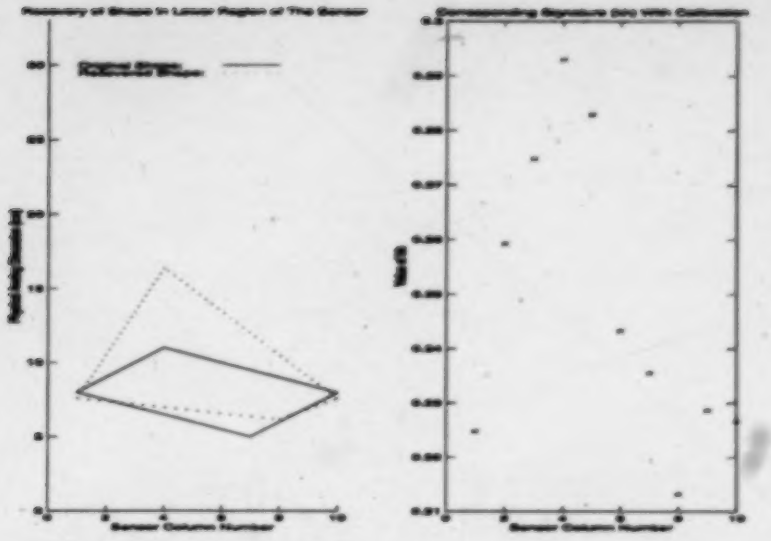


Figure 6.5: Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Lower Region of the CRS-CRS Sensor (with Calibration).

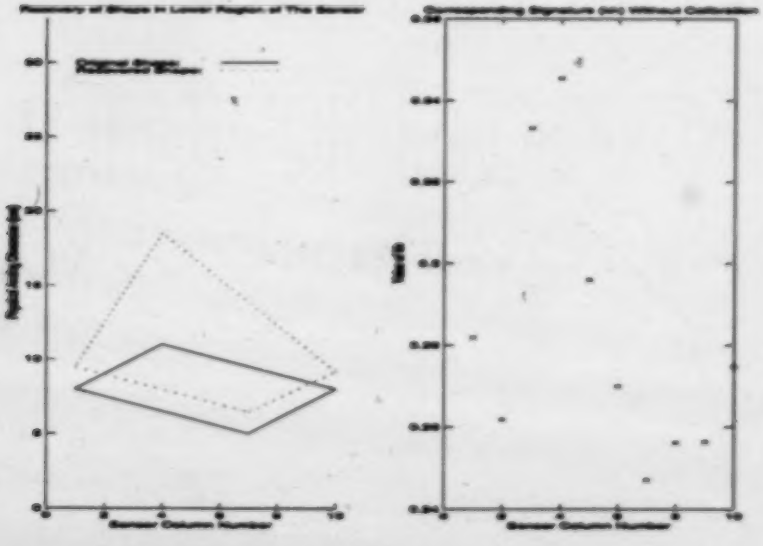


Figure 6.6: Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Lower Region of the CRS-CRS Sensor (without Calibration).

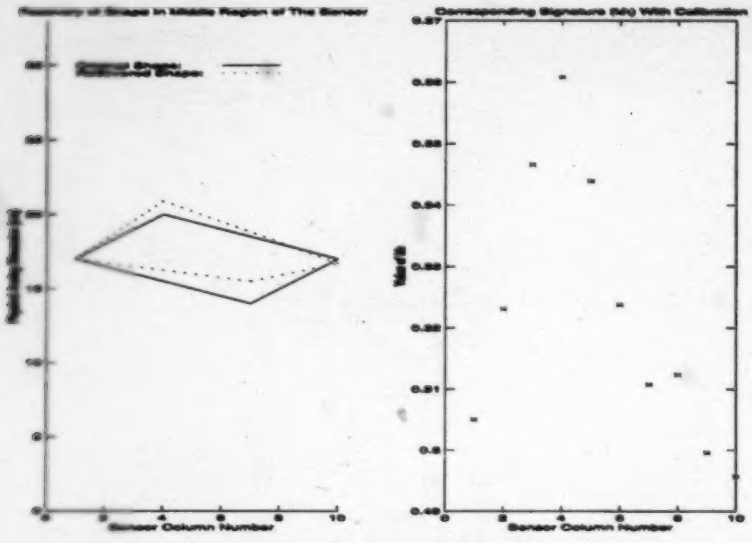


Figure 6.7: Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Middle Region of the CRS-CRS Sensor (with Calibration).

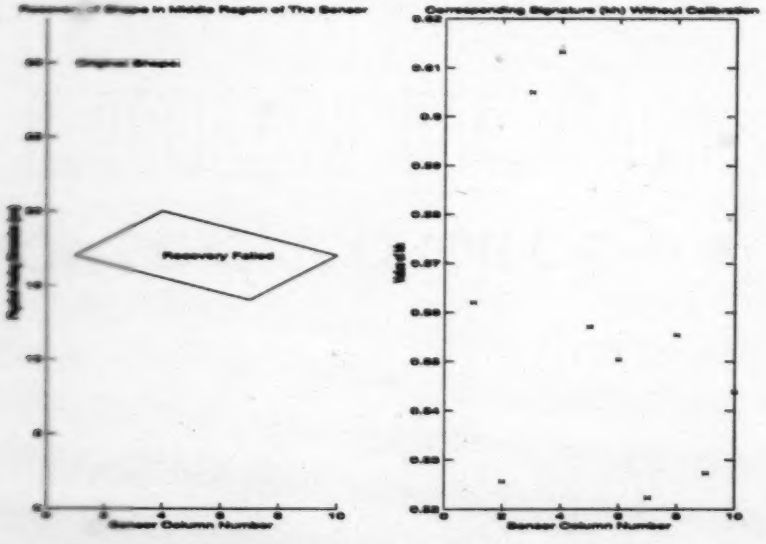


Figure 6.8: Noisy k_h-Signature of a Rectangle in the Middle Region of the CRS-CRS Sensor (without Calibration).

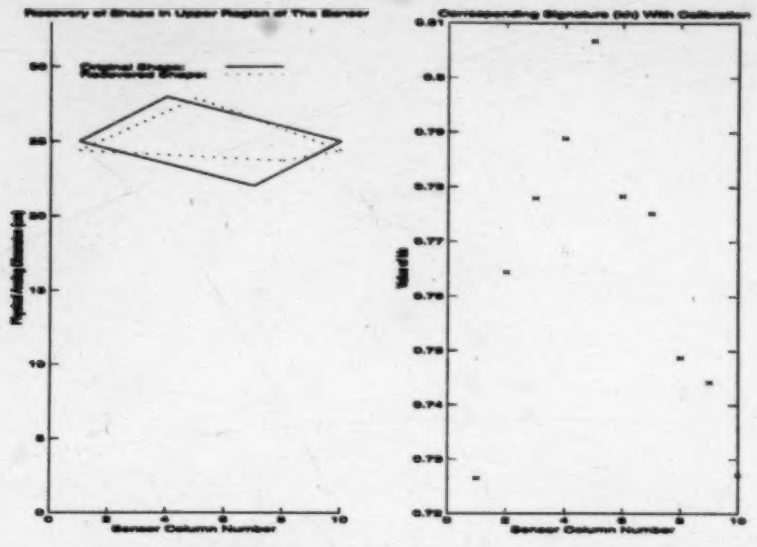


Figure 6.9: Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Upper Region of the CRS-CRS Sensor (with Calibration).

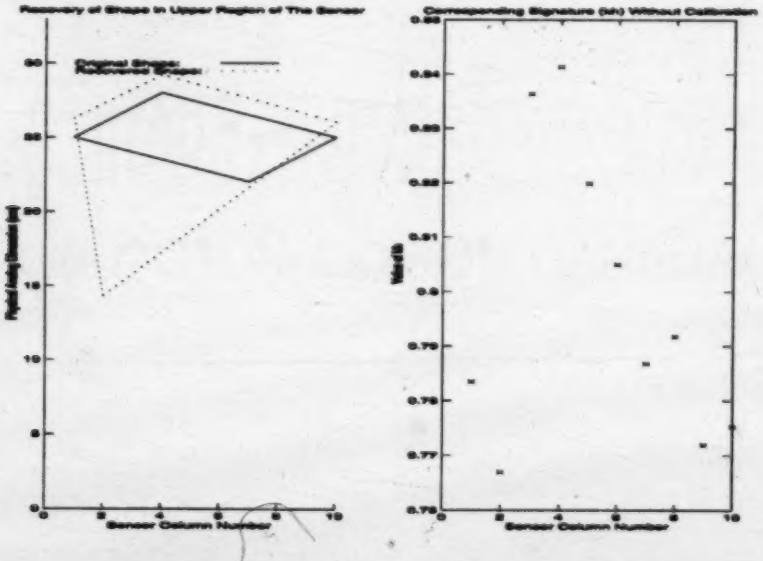


Figure 6.10: Recovering a Rectangle Using V.R. Algorithm with k_h-Signature in the Upper Region of the CRS-CRS Sensor (without Calibration).

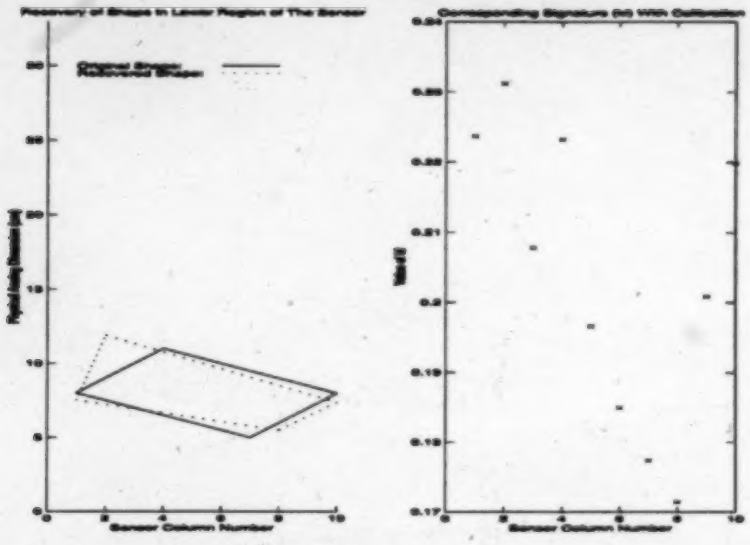


Figure 6.11: Recovering a Rectangle Using V.R. Algorithm with k_l-Signature in the Lower Region of the CRS-CRS Sensor (with Calibration).

6.1.3 Results of Recovering a Rectangle Using the Vertex Reconstruction Algorithm with a CRS-CRS Sensor and k_l-Pseudo-Signature

The values of k_l were also collected from the experiments described in the previous section. Vertex Reconstruction with k_l-signature was applied. Typical results are illustrated graphically in Figure 6.11 to Figure 6.16. In Figure 6.11, the object was put on the lower region of the CRS-CRS sensor. With calibration, the recovered shape was reasonably good. Figure 6.12 shows the corresponding un-calibrated signature, from which it was practically impossible to extract any transitional points. In Figure 6.13, the object was placed on the middle region of the sensor surface. The calibrated signature made it possible to recover a shape close to its original. In this case, calibration was vital and the un-calibrated signature was too noisy (see Figure 6.14). In Figure 6.15, the object was pressed against the upper region of the sensor. Due to relatively large error in the fourth sensing element, the recovered shape deviated quite a lot from the original. Figure 6.16 gives the seriously distorted signature without calibration. Table 6.3 lists the quantitative errors of the recovered shape.

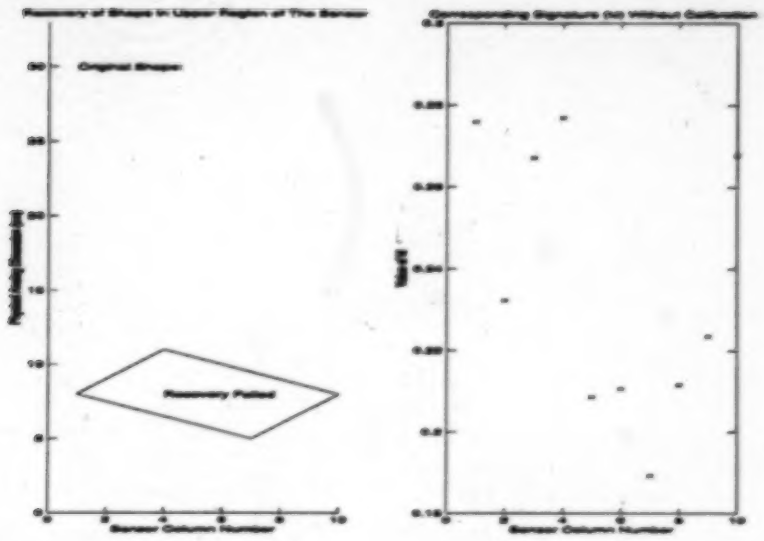


Figure 6.12: Noisy k_1-Signature of a Rectangle in the Lower Region of the CRS-CRS Sensor (without Calibration).

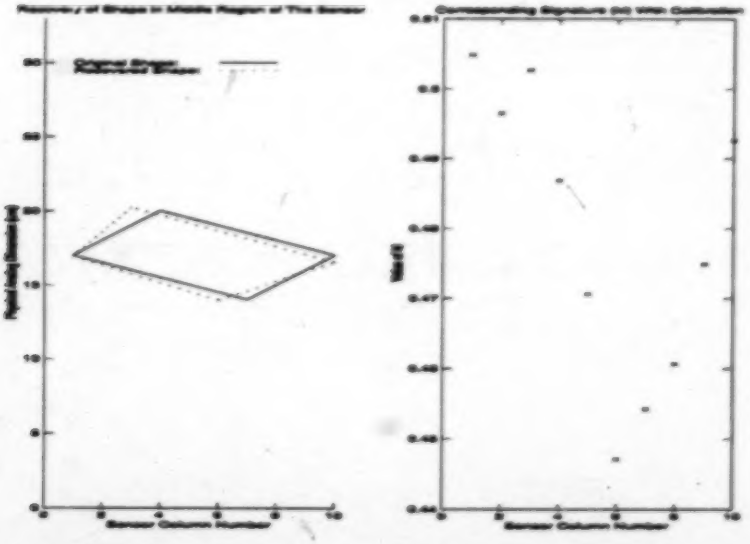


Figure 6.13: Recovering a Rectangle Using V.R. Algorithm with k_1-Signature in the Middle Region of the CRS-CRS Sensor (with Calibration).

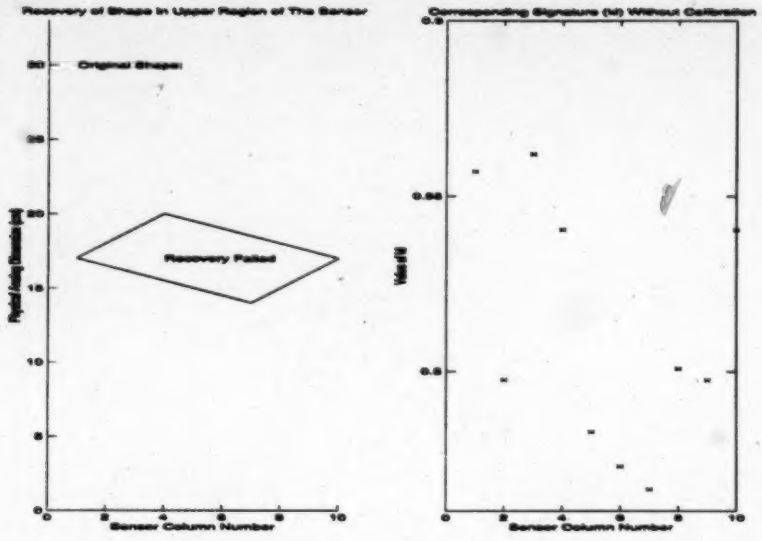


Figure 6.14: Noisy k_l-Signature of a Rectangle in the Middle Region of the CRS-CRS Sensor (without Calibration).

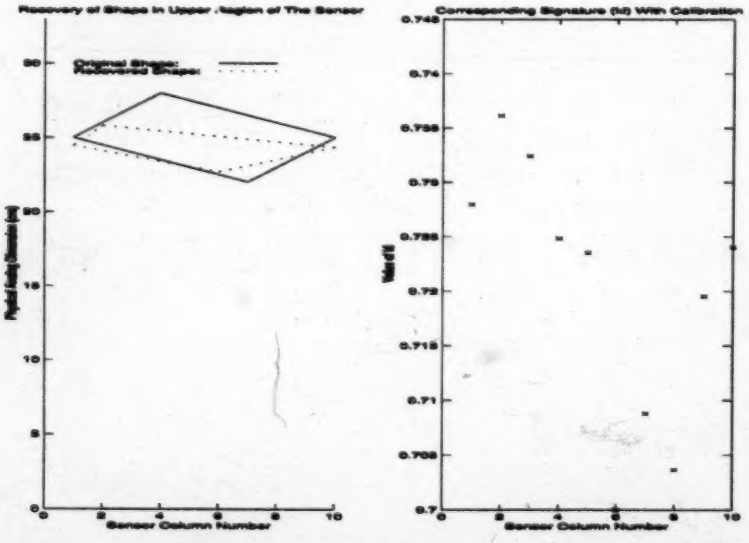


Figure 6.15: Recovering a Rectangle Using V.R. Algorithm with k_l-Signature in the Upper Region of the CRS-CRS Sensor (with Calibration).

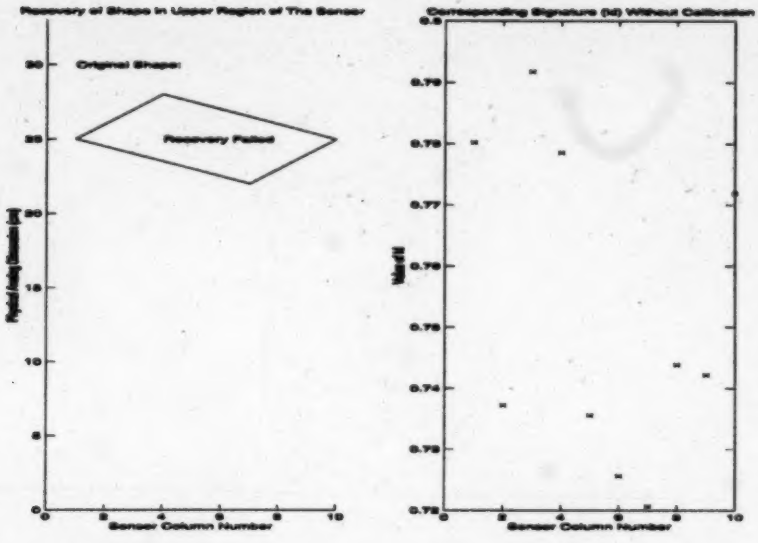


Figure 6.16: Noisy k_l-Signature of a Rectangle in the Upper Region of the CRS-CRS Sensor (without Calibration).

	Figure 6.11	Figure 6.13	Figure 6.15
M.C. (Radian-cm)	0.629	0.266	0.340
A.O.E. (Radian)	0.134	0.012	0.019
A.P.E. (cm)	0.261	0.502	0.656

Table 6.3: Quantitative Mismatch Between the Original and the Recovered Rectangle in Each Experiment (CRS-CRS Sensor Prototype and V.R. Algorithm with k_l-Signature).

6.1.4 Results of Recovering a Rectangle Using the Profile Reconstruction Algorithm with a CRS-CRS Sensor

Both k_h and k_l collected in the experiments described in Section 6.1.2 were fed to the Profile Reconstruction algorithm. Figure 6.17, Figure 6.19 and Figure 6.21 show the corresponding results of shape recovery with calibration. In all cases, the recovered profiles resembled the original closely. However, poor results were found when un-calibrated data were used for shape recovery, as shown in Figure 6.18, Figure 6.20 and Figure 6.22. Table 6.4 lists the quantitative errors in the recover shapes.

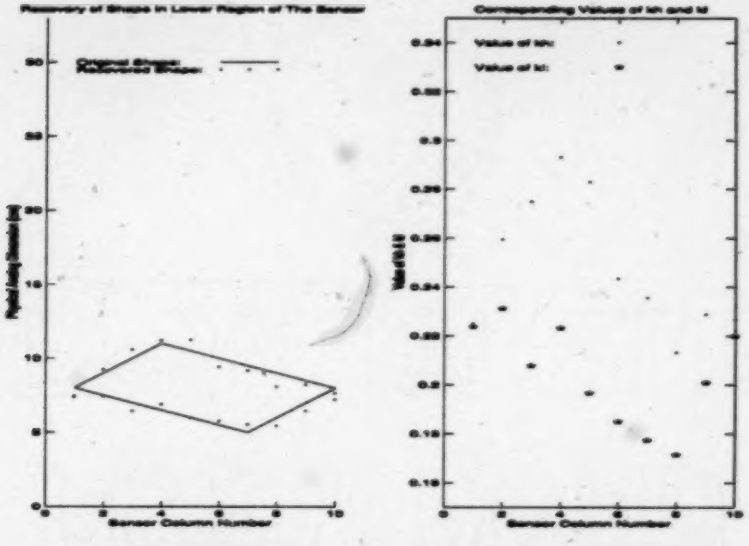


Figure 6.17: Recovering a Rectangle Using P.R. Algorithm in the Lower Region of the CRS-CRS Sensor (with Calibration).

	Figure 6.17	Figure 6.19	Figure 6.21
M.C. (Radian-cm)	0.397	0.223	0.273
A.O.E. (Radian)	0.094	0.011	0.006
A.P.E. (cm)	0.175	0.350	0.925

Table 6.4: Quantitative Mismatch Between the Original and the Recovered Rectangle in Each Experiment (CRS-CRS Sensor Prototype and P.R. Algorithm).

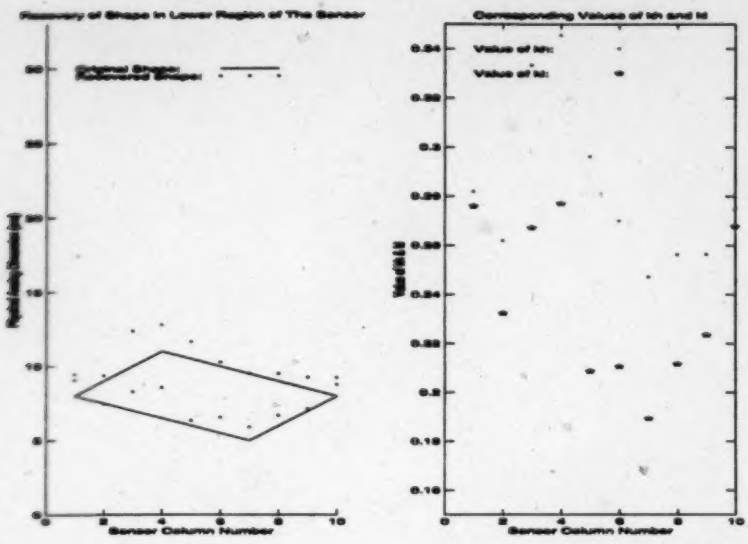


Figure 6.18: Recovering a Rectangle Using P.R. Algorithm in the Lower Region of the CRS-CRS Sensor (without Calibration).

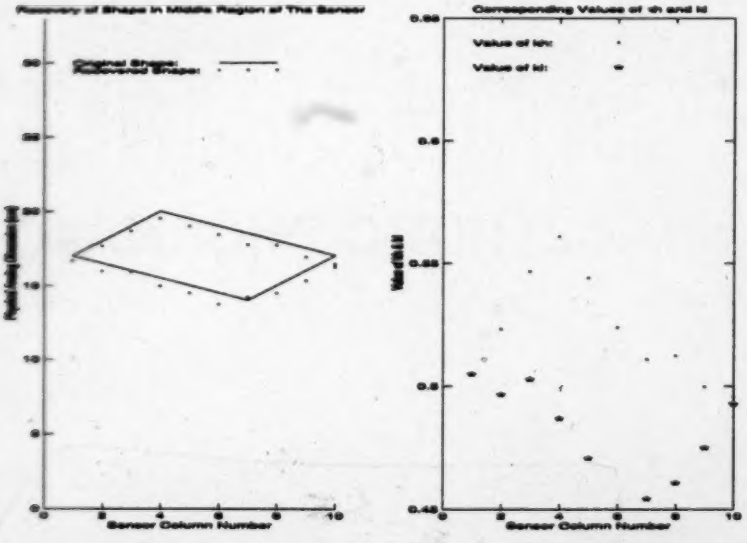


Figure 6.19: Recovering a Rectangle Using P.R. Algorithm in the Middle Region of the CRS-CRS Sensor (with Calibration).

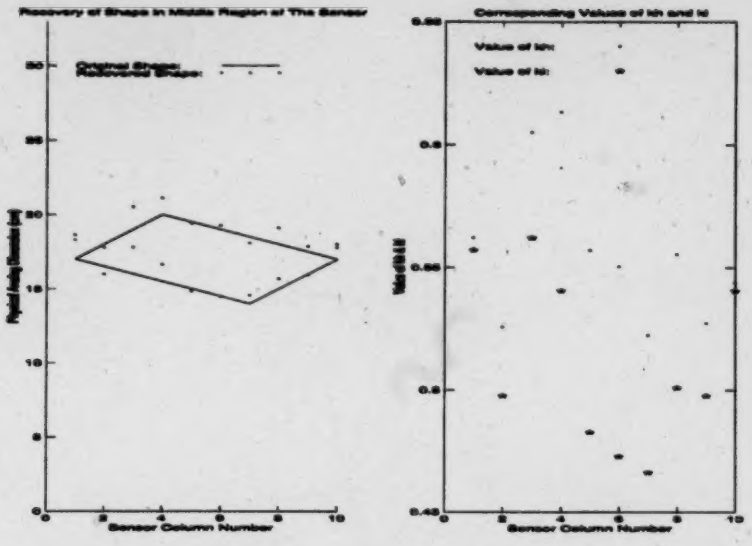


Figure 6.20: Recovering a Rectangle Using P.R. Algorithm in the Middle Region of the CRS-CRS Sensor (without Calibration).

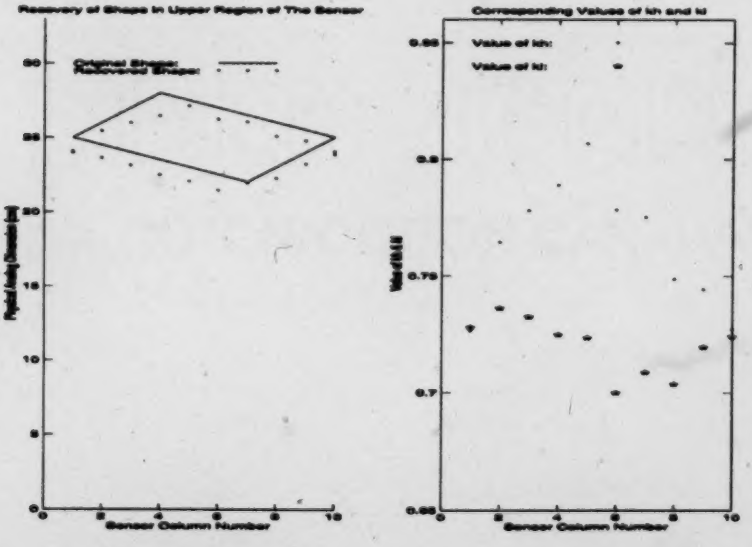


Figure 6.21: Recovering a Rectangle Using P.R. Algorithm in the Upper Region of the CRS-CRS Sensor (with Calibration).

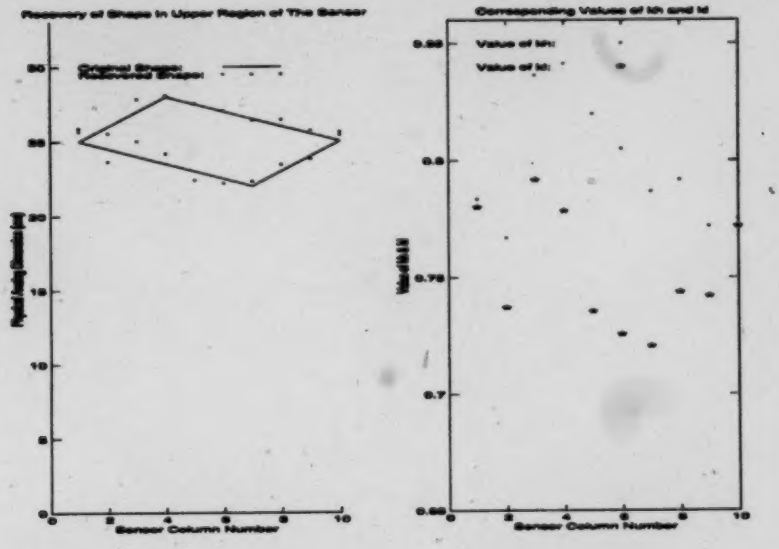


Figure 6.22: Recovering a Rectangle Using P.R. Algorithm in the Upper Region of the CRS-CRS Sensor (without Calibration).

6.2 Experiments with a CRS-M Sensor

The prototype of a CRS-M hybrid sensor is shown in Figure 6.23. It measured 4cm (digital dimension) by 25cm (analog dimension). The same conductive silicone rubber as in the CRS-CRS prototype was used. Strips with 3 mm in width were sliced from the rubber sheet. Nine strips were aligned in parallel on top of a bare clad board. Non-conductive fabric mesh was laid between layer of strips on top and the conductive copper surface at the bottom. When no force was applied, the rubber strips did not touch the metal surface of the clad board. Since there were only nine columns of sensing elements in the setup, the resolution in the digital dimension was limited to 2.25 sensing elements per cm. With visual measurement using a tape measure, the resolution in the analog dimension was 0.1 cm. This is equivalent to a maximum C.Q.E. of $\pm\frac{1}{250}$. For simplicity, only a rectangular object was used in the experiments.

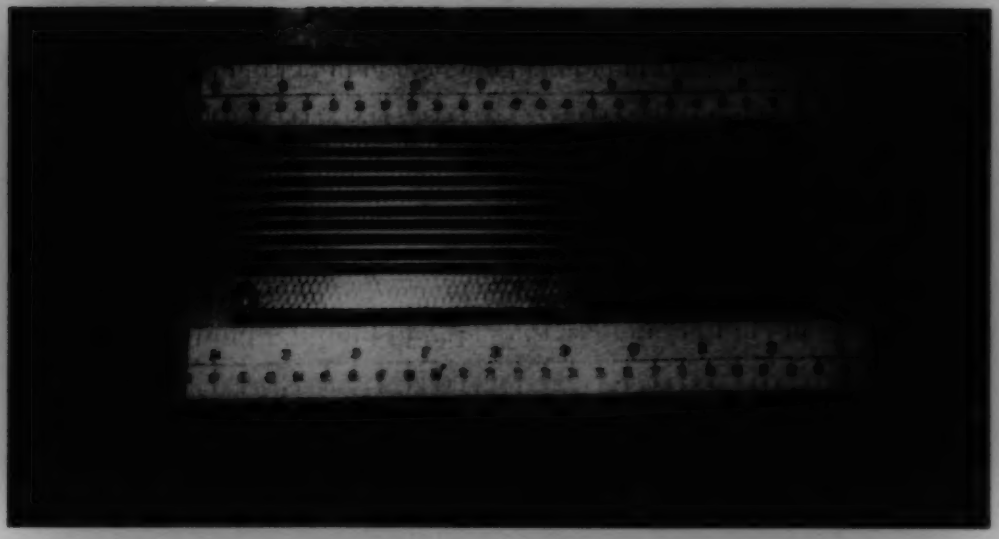


Figure 6.23: Prototype of a CRS-M Sensor with Nine Sensing Elements

6.2.1 Calibrations of a CRS-M Sensor

Calibration procedure similar to that in a CRS-CRS sensor was done. Since there was just a set of V_{out} in case of a CRS-M Sensor, only single-measurement was required. The steps were as follows:

1. A point contact was made at the 1, 2, 3, ... unit mark in turn. Corresponding V_{out} were recorded. They were compared to the expected values according to equation (3.7). The mean was then calculated as $V_{out(w=0)}$.
2. A 5-unit long contact was made starting at the 1, 2, 3, ... unit marks. Another mean, $V_{out(w=5)}$ was obtained.
3. Repeat step 2 above with a 10-unit length of contact and obtain the mean, $V_{out(w=10)}$.

Finally, the calibration factors were calculated as follows:

$$V_{outOffset} = \frac{V_{out(w=0)} + V_{out(w=5)} + V_{out(w=10)}}{3} \tag{6.5}$$

Figure 6.24 to Figure 6.26 show the corresponding calibration graphs of a sensing element in a CRS-M Sensor Prototype. Table 6.5 tabulates the calibration factors for each of the sensing element. These calibration factors are used in a similar way as in the CRS-CRS sensor. It is a compromised approximation of a non-linear function by a linear function, but it works reasonably well with the CRS-M sensor prototype.

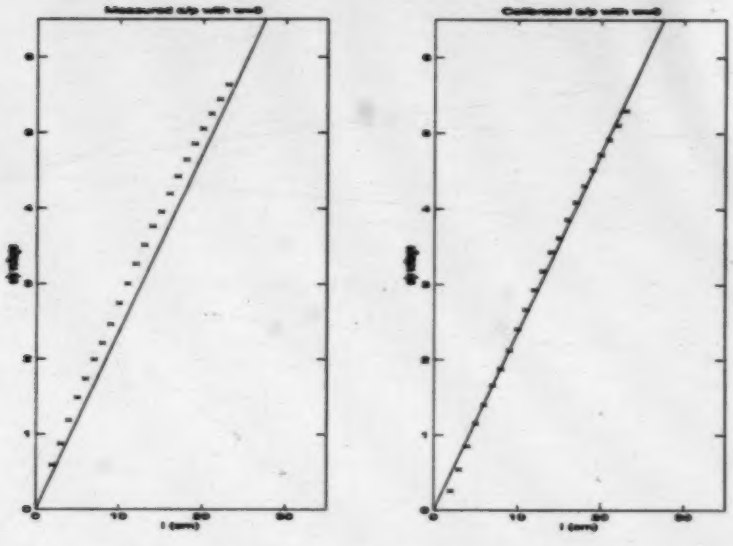


Figure 6.24: Calibration Graphs for the First Sensing Element in the CRS-M Hybrid Sensor Prototype ($w = 0$).

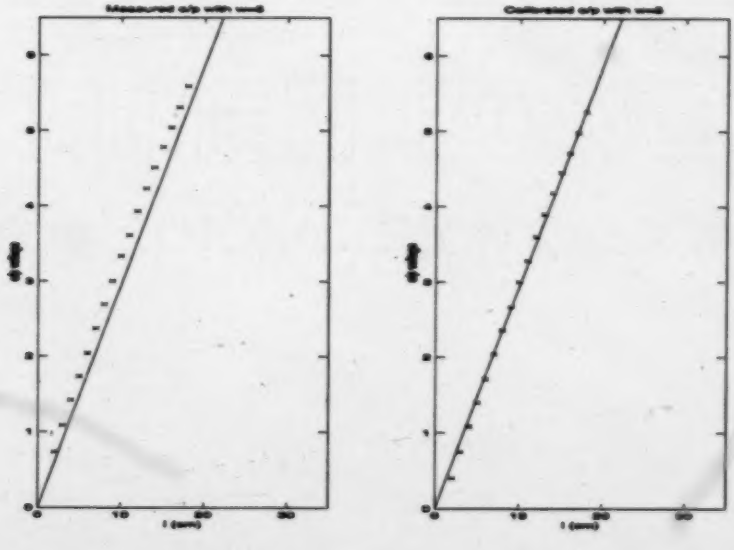


Figure 6.25: Calibration Graphs for the First Sensing Element in the CRS-M Hybrid Sensor Prototype ($w = 5$).

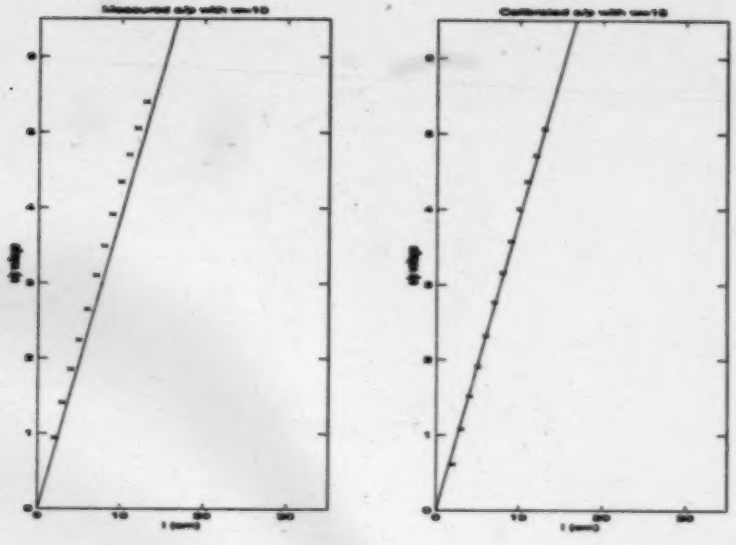


Figure 6.26: Calibration Graphs for the First Sensing Element in the CRS-M Hybrid Sensor Prototype ($w = 10$).

$V_{ref} = 6.05$V	Sensing Element Number								
	1	2	3	4	5	6	7	8	9
Calibration Factor for V_{out} (V)	0.333	0.175	0.068	0.304	0.201	0.199	0.009	0.024	0.087

Table 6.5: Calibration Factors for Each Sensing Element of the CRS-M Hybrid Sensor Prototype.

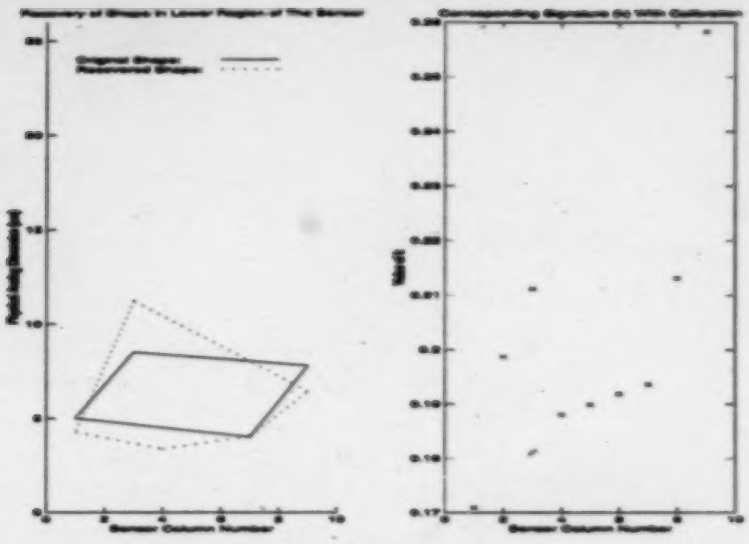


Figure 6.27: Recovering a Rectangle Using V.R. Algorithm in the Lower Region of the CRS-M Sensor (with Calibration).

6.2.2 Results of Recovering a Rectangle Using the Vertex Reconstruction Algorithm with a CRS-M Sensor

Experiments were done with a rectangular planar object of dimension 5 cm by 3.5 cm. Typical results are illustrated graphically in Figure 6.27 to Figure 6.32. In Figure 6.27, the object was placed near the lower region of the CRS-M sensor. Transitions were detected easily. However, errors in k caused relative large discrepancies in estimating the vertices. Figure 6.30 shows that without calibration, the errors in the signature were even worse. In Figure 6.29, the object was located on the middle region of the sensor surface. Accurate transitions were detected and the recovered shape resembled the original closely. Figure 6.30 shows the result of using an un-calibrated signature. In Figure 6.31, the object was put on the upper region of the sensor. Prominent transitions could be identified from the pseudo-signature. However, due to noise in the value of k from the seventh sensing element, relatively large error was resulted in reconstructing the lower right vertex of the rectangular object. For completeness, Figure 6.32 shows the result of shape recovery using an uncalibrated signature. Table 6.6 shows the quantitative errors in shape recovery.

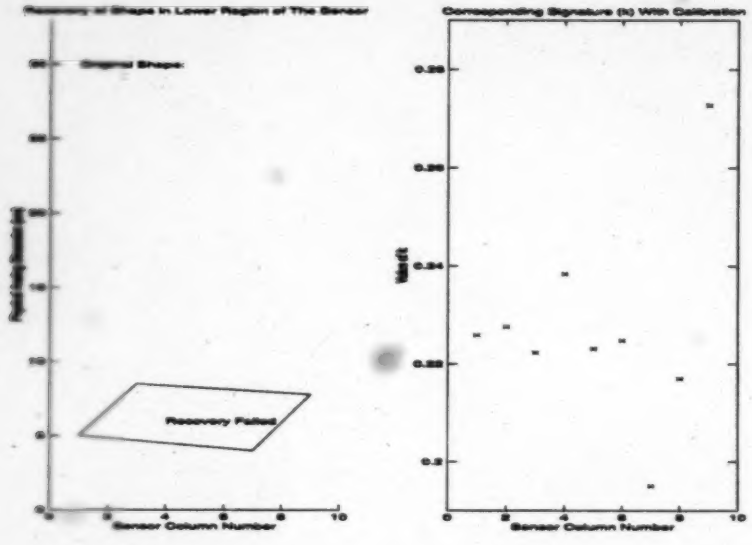


Figure 6.28: Noisy k_l-Signature of a Rectangle in the Lower Region of the CRS-M Sensor (without Calibration).

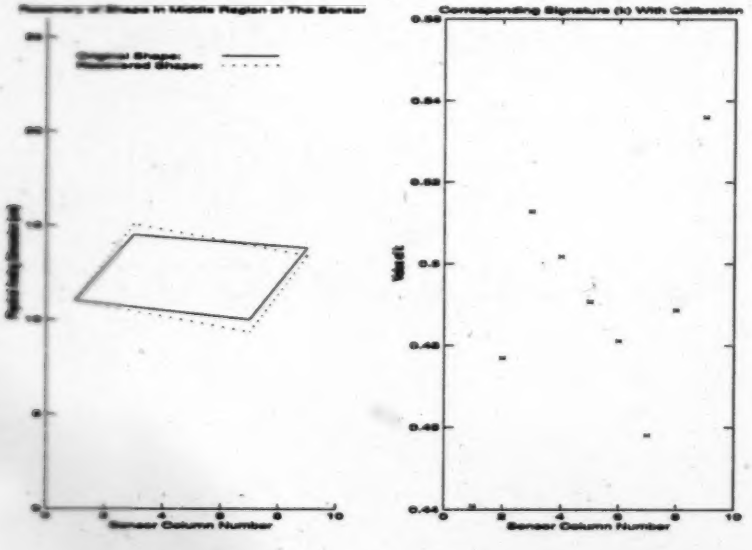


Figure 6.29: Recovering a Rectangle Using V.R. Algorithm in the Middle Region of the CRS-M Sensor (with Calibration).

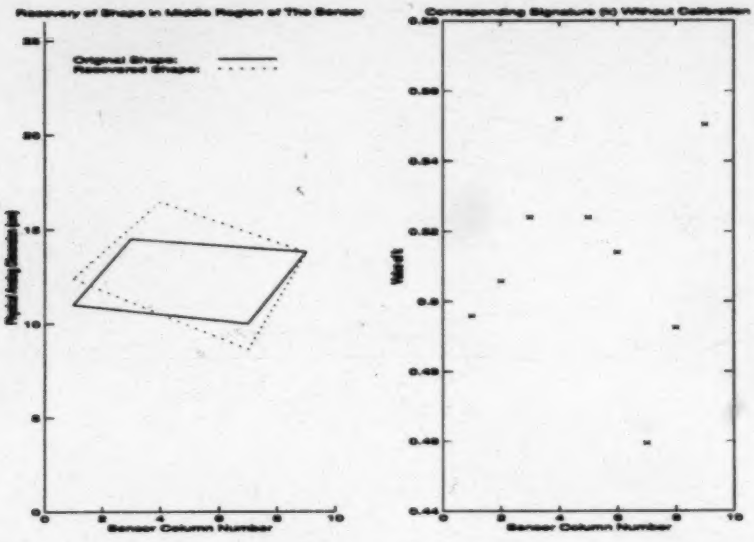


Figure 6.30: Recovering a Rectangle Using V.R. Algorithm in the Middle Region of the CRS-M Sensor (without Calibration).

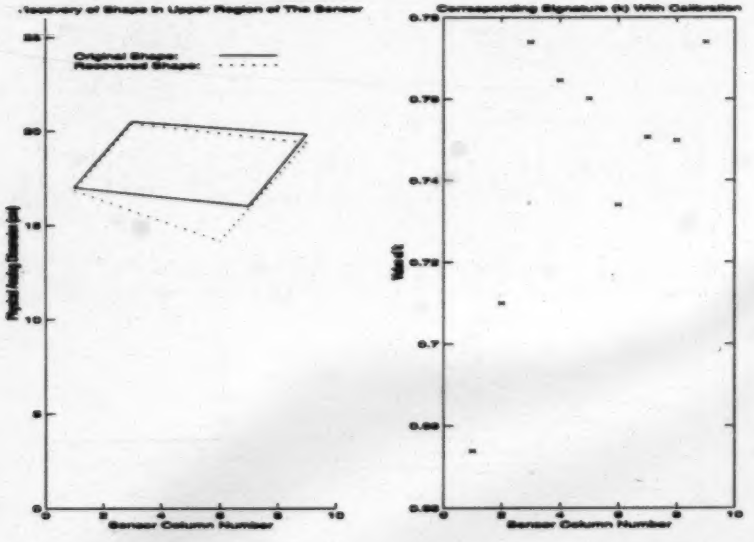


Figure 6.31: Recovering a Rectangle Using V.R. Algorithm in the Upper Region of the CRS-M Sensor (with Calibration).

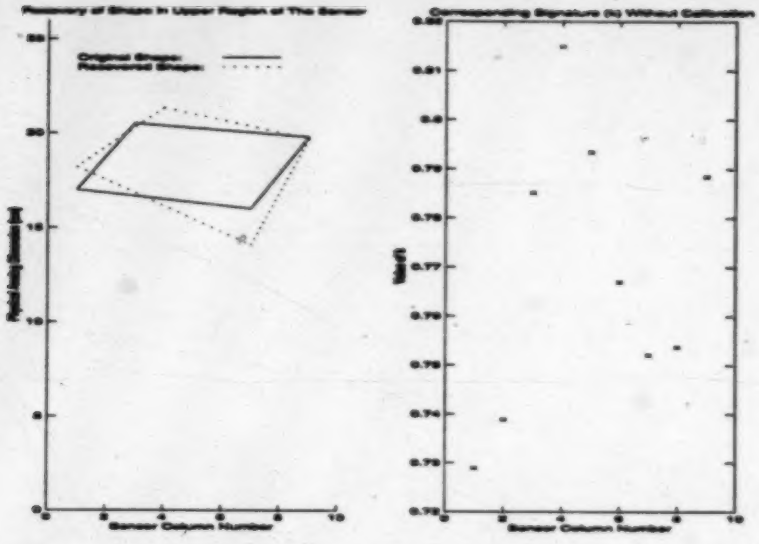


Figure 6.32: Recovering a Rectangle Using V.R. Algorithm in the Upper Region of the CRS-M Sensor (without Calibration).

	Figure 6.27	Figure 6.29	Figure 6.30	Figure 6.31	Figure 6.32
M.C. (Radian-cm)	0.930	0.132	0.390	0.358	0.535
A.O.E. (Radian)	0.050	0.028	0.226	0.084	0.290
A.P.E. (cm)	0.475	0.125	0.707	0.673	0.250

Table 6.6: Quantitative Mismatch Between the Original and the Recovered Rectangle in Each Experiment (CRS-M Sensor Prototype and V.R. Algorithm with k-Signature).

6.3 Experiments with a Cyl-CRS-M Sensor

The prototype of a Cyl-CRS-M hybrid sensor is shown in Figure 6.33. It measured 10 cm in length by 1.8 cm in diameter. The same conductive silicone rubber as in the CRS-M prototype was used. A section of PVC pipe was used for the cylindrical base structure. The metallic bottom layer was a sheet of aluminum foil coated on the PVC pipe surface. Then a fabric mesh was laid on the metallic surface to serve as a separator. Finally, strips with 3 mm in width were sliced from the rubber sheet. Five of these strips were aligned in parallel on top of the fabric mesh. Electrodes were connected to both ends of each rubber strip. To protect the sensing surface, a tough PVC plastic sheet was wrapped around the whole structure. When no force was applied, the rubber strips did not touch the aluminum foil. Since there were only five columns of sensing elements in the setup, the resolution in the

Figure 6.33: Prototype of a Cyl-CRS-M Sensor with Nine Sensing Elements

digital dimension was limited to 2.5 sensing elements per cm on a radius of curvature of 0.9 cm. As measurement was done visually using a tape measure, the resolution in the analog dimension was 0.1 cm. This is equivalent to a maximum C.Q.E. of $\pm\frac{1}{100}$. Only point and line contacts were tested with this prototype.

6.3.1 Calibrations of a Cyl-CRS-M Sensor

As the Cyl-CRS-M Sensor was used to detect point and line contacts, calibration with $w = 0$ was sufficient. Figure 6.34 shows the corresponding calibration graphs of a sensing element in a CRS-M Sensor Prototype and Table 6.7 lists the calibration factors for each of the sensing elements.

$V_{ref} = 6.05$V	Sensing Element Number				
	1	2	3	4	5
Calibration Factor for V_{out} (V)	0.345	0.016	0.353	0.317	0.080

Table 6.7: Calibration Factors for Each Sensing Element of the Cyl-CRS-M Hybrid Sensor Prototype.

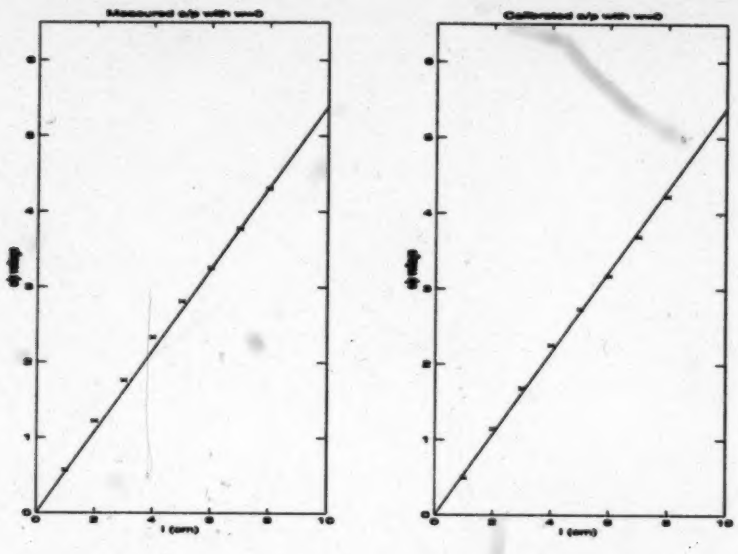


Figure 6.34: Calibration Graphs for the First Sensing Element in the Cyl-CRS-M Hybrid Sensor Prototype ($w = 0$).

6.3.2 Results of Recovering Points/Lines Contacts with a Cly-CRS-M Sensor

Due to curvature of the cylindrical sensing surface of the prototype, any edge of a contact object actually touched two of the sensing elements at most. Any line-contact situation was practically resolved into two-point-contact scenario. Five sets of two-point contacts were made on the Cly-CRS-M sensor and Figure 6.35 shows the results of the experiments. In general, the location of any point contact was extracted from the calibration graph and the recovered points were closely matched to the originals.

6.4 Summary of Experiments

A 12-bit precision A/D conversion is common in practice which means a maximum of $\pm\frac{1}{8192}$ part of error might occur in the output voltage conversion process. This is insignificant as compared to the quantization error due to the length of contact between an object and the hybrid sensor, for instance $\pm\frac{1}{325}$, $\pm\frac{1}{250}$ and $\pm\frac{1}{100}$ in the CRS-CRS, CRS-M and Cyl-CRS-M sensor prototype respectively. Though they were larger than the value of $\pm\frac{1}{1000}$ used in the simulations, no significant impact was found from the experiments. Experiments also

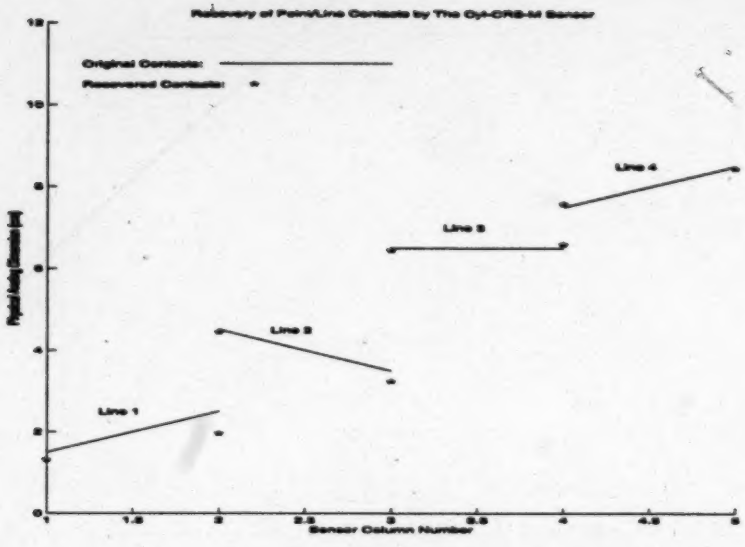


Figure 6.35: Recovering Points/Lines Contacts Using the CRS-M Sensor.

revealed that after calibration was done on the sensor outputs, the errors were similar to that of P.U. profile II used in the simulations. Table 6.8 to Table 6.11 show the corresponding errors of each sensing elements after calibration. The error of V_h w.r.t. V_{ref} in a CRS-CRS sensor was approximately -1.0% to +1.1%. With the same CRS-CRS sensor, the error of V_l w.r.t. V_{ref} was -1.2% to +1.1%. For the CRS-M sensor, the error of V_{out} w.r.t. V_{ref} ranged from -1.1% to +1.0%. For the Cyl-CRS-M sensor, the error of V_{out} w.r.t. V_{ref} was approximately -0.9% to +0.4%. In conclusion, the performance of shape recovery in the experiments matches that in the simulation with a P.U. profile II.

Reference	Sensing Element Number									
	1	2	3	4	5	6	7	8	9	10
Figure 6.5	1.1%	0.9%	1.0%	1.0%	0.9%	0.5%	0.3%	0.4%	1.0%	1.1%
Figure 6.7	0.1%	0.1%	0.3%	0.3%	-0.7%	-1.0%	-0.7%	0.7%	-1.0%	0.7%
Figure 6.9	0.3%	-0.7%	0.4%	-1.0%	-0.9%	-0.1%	-1.0%	-0.8%	-0.1%	0.2%

Table 6.8: Percentage Errors in V_h w.r.t. V_{ref} for Each Sensing Element of the CRS-CRS Hybrid Sensor Prototype in the Experiments (after Calibration).

Reference	Sensing Element Number									
	1	2	3	4	5	6	7	8	9	10
Figure 6.11	0.8%	1.2%	0.3%	0.7%	1.0%	1.1%	0.7%	1.0%	0.5%	1.1%
Figure 6.13	0.1%	-0.1%	0.2%	1.1%	-0.5%	-1.2%	-1.1%	0.9%	-1.0%	1.0%
Figure 6.15	0.1%	-0.7%	0.3%	-1.1%	-0.3%	-0.4%	-0.8%	-0.5%	-0.7%	0.5%

Table 6.9: Percentage Errors in V_i w.r.t. V_{ref} for Each Sensing Element of the CRS-CRS Hybrid Sensor Prototype in the Experiments (after Calibration).

Reference	Sensing Element Number								
	1	2	3	4	5	6	7	8	9
Figure 6.27	1.0%	0.7%	0.3%	0.5%	0.6%	0.7%	1.0%	0.4%	0.5%
Figure 6.29	0.8%	0.1%	0.2%	0.4%	0.3%	0.3%	-0.9%	-0.3%	-0.4%
Figure 6.31	0.4%	1.0%	-0.2%	-0.5%	-0.2%	-0.3%	-1.1%	-0.7%	-0.9%

Table 6.10: Percentage Errors in V_{out} w.r.t. V_{ref} for Each Sensing Element of the CRS-M Hybrid Sensor Prototype in the Experiments (after Calibration).

Reference in Figure 6.35	Sensing Element Number				
	1	2	3	4	5
Line 1	-0.5%	-0.9%	-	-	-
Line 2	-	-0.4%	-0.6%	-	-
Line 3	-	-	-0.3%	0.4%	-
Line 4	-	-	-	0.3%	-0.4%

Table 6.11: Percentage Errors in V_{out} w.r.t. V_{ref} for Each Sensing Element of the Cyl-CRS-M Hybrid Sensor Prototype in the Experiments (after Calibration).

Chapter 7

Conclusions

This thesis presented a new tactile sensor design, called hybrid tactile sensor, and described the two shape interpretation algorithms, the Vertex Reconstruction Algorithm and the Profile Reconstruction algorithm, which reconstruct the contact object's shape from sensor outputs. Four hybrid tactile sensor models, namely, the CRS-CRS, the CRS-M, the CRS-CRP and the Cyl-CRS-M were presented. They were studied in detail, except for the CRS-CRP design which was too complicated for a convenient analytical model. Based on the mathematical models of the sensor designs, simulations were conducted for verification and performance estimation. Three prototypes were constructed using inexpensive materials. Experiments were carried out with each prototype and their results were analyzed and compared.

7.1 Summary of Objectives

This thesis categorized sensor technologies into analog sensing and digital sensing. Analog sensing technology provided a simple and inexpensive means to abstract tactile information. However, it was limited in giving point contact data only. On the other hand, digital sensing technology was capable of giving more detail such as the shape and orientation of an object in touch with the sensing surface. Yet, it involved more complex circuitry, larger amount of raw data and slower sampling operations as compared to its analog counterpart. This research focused on a trade-off between the above two extremes and attempted a new hybrid tactile sensor design with the following advantages:

1. The tactile sensor should be simple in structure and inexpensive to fabricate.
2. The tactile sensor should be able to provide information on the location and the shape of contact.

3. The tactile sensor should be flexible enough to conform to different geometric shapes.

7.2 Summary of Simulations

Simulations were done to verify the behavior of each hybrid sensor model. In addition, the validity of each shape reconstruction algorithm was checked with different error levels in sensor outputs. Polygonal shapes such as triangles, tetragons and hexagons were used in the simulations. Different error conditions were assumed in each simulation set. These conditions include:

1. Ideal Sensor Output (Noise-free Sensor).
2. Sensor Output with C.Q.E. (and T.E., if applicable) Only.
3. Sensor Output with C.Q.E. (and T.E., if applicable) and P.U. profile I.
4. Sensor Output with C.Q.E. (and T.E., if applicable) and P.U. profile II.

It was found that each hybrid sensor model and the applicable shape reconstruction algorithms worked perfectly under ideal conditions. Even with C.Q.E. (and T.E. in case of using the Vertex Reconstruction Algorithm), shape recovery was reasonably good. Two profiles of P.U. were introduced on top of C.Q.E. (and T.E.), the performance of both shape interpretation algorithm deteriorated. P.U. profile I approximated the characteristics of the prototype closely and still produced reasonably good results. P.U. profile II caused much noisy conditions and the algorithms performed poorly or even failed to work.

7.3 Summary of Prototypes and Experiments

As shape reconstruction algorithms were sensitive to sensor output errors, it was necessary to see whether low sensor output errors could be achieved in practice. Three sensor prototypes were made in the research. The CRS-CRS sensor and the CRS-M sensor were two simple planar structures, whereas the Cyl-CRS-M was a cylindrical one. The elastic conductive rubber strips of these sensors were prepared from conductive silicone rubber sheets. These conductive silicone rubber sheets were carbon-based conductive material widely used in RFI/EMI shielding. It was readily available in various resistivities and thickness at very low cost. Besides, rubber strips made from these conductive silicone rubber exhibited extremely good linearity in resistivities. For insulation between the two structural layers of a hybrid sensor, fabric meshes that are easily found in a fabric store were used as separators.

Experiments on shape recovery were done with each prototype and the applicable shape reconstruction algorithms. Since the resolutions in the digital dimensions of the prototypes were small, only a simple rectangular shape was used for the experiments. Calibrations were required to obtain the necessary offset voltage to correct the sensor outputs. From the results of the experiments, the shape reconstruction algorithms gave reasonable shape recovery.

7.4 Summary of Achievement

The new hybrid tactile sensor models combined the advantages of both analog and digital sensing technologies and the following is achieved in the research:

1. The sensor was simple because only n or $2n$ sampling operations are required to cover an area that normally requires n^2 sampling operations. This represents a considerable saving in sampling time especially when n is large.
2. It was shown that very inexpensive materials can be used for fabrication of the hybrid sensors.
3. The flexibility of the materials and the overall sensor structure made it possible for a hybrid sensor to fit into different geometric surfaces.
4. Due to simplicity of a typical hybrid sensor, even sensors with large physical dimensions could be manufactured at relatively low cost.
5. No special circuit was required to eliminate the parallel path problem in a conventional digital sensor.

Through simulation and experiment with the sensor prototypes, the two accompanying shape recovery algorithms were verified. Their behavior under noisy conditions were compared and could serve as guidelines in choosing materials and sensor models in achieving different quality requirement. The sensor prototypes used in the experiments were not in production-quality, but the results in shape recovery were still satisfactory. Since much more precise dimensions and control of materials can be achieved in industrial production process (for instance, resistors with 0.1% tolerance are commercially available), hybrid sensor with lower noise level than the prototypes can easily be made. The hybrid tactile sensor designs discussed in this thesis provide simple and economical alternatives in tactile sensor.

7.5 Further Research

In this thesis, new hybrid tactile sensor designs and the accompanying shape recovery algorithms were presented. Its simplicity provided a new alternative in tactile sensor technologies. Further research on these designs may involve the following:

1. Improvement on the physical construction and materials of the prototypes to minimize error levels of the sensor outputs.
2. Improvement on transition detection in a pseudo-signature when applying the Vertex Reconstruction algorithm for shape recovery.
3. Investigation on the performance of the hybrid sensor model with different geometric structure, for instance, semi-spherical shape such as a finger tip.
4. Investigation on the possibility of shape recovery of multiple objects with overlapping pseudo-signatures.

Bibliography

[1] A. Namiki and M. Ishikawa. Optimal grasping using visual and tactile feedback. In *Proceedings of 1996 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems*, pages 584–596, 1996.

[2] M. Charlebois, K. Gupta, and S. Payandeh. Shape description of general, curved surfaces using tactile sensing and surface normal information. In *Proceedings of 1997 IEEE International Conference on Robotics and Automation*, pages 2819–2824, 1997.

[3] M. Hakozaki, H. Oasa, and H. Shinoda. Telemetric robot skin. In *Proceedings of 1999 IEEE International Conference on Robotics and Automation*, pages 957–961, 1999.

[4] Howard R. Nicholls and Mark H. Lee. A survey of robot tactile sensing technology. *The International Journal of Robotics Research*, 8(3):3–30, June 1989.

[5] B. E. Robertson and A. J. Walkden. Tactile sensor system for robotics. In *Robot Sensors, Vol 2: Tactile and Non-Vision*, pages 89–97. IFS (Publications) Ltd, UK, 1986.

[6] M. Inaba, Y. Hoshino, K. Nagasaka, T. Ninomiya, S. Kagami, and H. Inoue. A full-body tactile sensor suit using electrically conductive fabric and strings. In *Proceedings of 1996 IEEE International Conference on Intelligent Robots and Systems*, pages 450–457, 1996.

[7] P. Dario and D. De Rossi. Tactile sensors and the gripping challenge. *IEEE Spectrum*, pages 46–52, August 1985.

[8] J. S. Son, E. A. Monteverde, and R. D. Howe. A tactile sensor for localizing transient events in manipulation. In *Proceedings of 1994 IEEE International Conference on Robotics and Automation*, pages 471–476, 1994.

[9] J. Dargahi, M. Parameswaran, and S. Payandeh. A micromachined piezoelectric tactile sensor for use in endoscopic graspers. In *Proceedings of 1998 IEEE International Conference on Intelligent Robots and Systems*, pages 1503–1508, 1998.

[10] D. M. Siegel, S. M. Drucker, and I. Garabicta. Performance analysis of a tactile sensor. In *Proceedings of 1987 IEEE International Conference on Robotics and Automations*, pages 1493–1499, 1987.

[11] D. De Rossi, A. Nannini, and C. Domenici. Artificial sensing skin mimicking mechanoelectrical conversion properties of human dermis. *IEEE Transactions on Biomedical Engineering*, 35(2):83–92, February 1988.

[12] H. Shinoda, K. Matsumoto, and S. Ando. Acoustic resonant tensor cell for tactile sensing. In *Proceedings of 1997 IEEE International Conference on Robotics and Automation*, pages 3087–3092, 1997.

[13] B. L. Hutchings, A. R. Grahn, and R. J. Petersen. Multiple-layer cross-field ultrasonic tactile sensor. In *Proceedings of 1994 IEEE International Conference on Robotics and Automations*, pages 2522–2528, 1994.

[14] Bert Tise. A compact high resolution piezoresistive digital tactile sensor. In *Proceedings of 1994 IEEE International Conference on Robotics and Automations*, pages 760–764, 1988.

[15] Carroll Touch. *Touch Handbook*. Carroll Touch, P.O. Box 1309, Round Rock, Texas 78680, 1996.

[16] L. D. Harmon. Automated tactile sensing. *The International Journal of Robotics Research*, 1(2):3–31, 1982.

[17] Esther M. Arkin, L. Paul Chew, Daniel P. Huttenlocher, Klara Kedem, and Joseph S. B. Mitchell. An efficiently computable metric for comparing polygonal shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 13(3):209–216, March 1991.

[18] Fujipoly. *Electronic Packaging Components*. Fujipoly, 365 Carnegie Avenue, P.O. Box 679, Kenilworth, NJ 07033-0679, USA, 1996.